



**HAL**  
open science

# Une algèbre pour la représentation et la manipulation spatio-temporelle et sémantique des trajectoires d'objets mobiles

Donia Zheni Épouse Triki

► **To cite this version:**

Donia Zheni Épouse Triki. Une algèbre pour la représentation et la manipulation spatio-temporelle et sémantique des trajectoires d'objets mobiles. Informatique [cs]. Université de Tunis El Manar, 2015. Français. NNT: . tel-01275831

**HAL Id: tel-01275831**

**<https://hal.science/tel-01275831>**

Submitted on 18 Feb 2016

**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.



UNIVERSITÉ DE TUNIS EL MANAR  
FACULTÉ DES SCIENCES DE TUNIS

---

Ecole Doctorale de Mathématiques, Informatique,  
Sciences et Technologies des Matériaux

## THÈSE

présentée en vue de l'obtention du

# Diplôme de Docteur en Informatique

par

**Donia ZHENI**

(Mastère en Informatique, FST)

---

---

**Une algèbre pour la représentation et la manipulation  
spatio-temporelle et sémantique des trajectoires d'objets  
mobiles**

---

Soutenue le 22 Septembre 2015 devant le jury d'examen

Pr. Khaled BSAIES	FST, Tunisie	Président
Pr. Thomas DEVOGELE	Université de Tours, France	Rapporteur
Pr. Alain BOUJU	Université de La Rochelle, France	Rapporteur
Pr. Faouzi MOUSSA	FST, Tunisie	Examineur
Pr. Henda BEN GHEZALA	ENSI, Tunisie	Directeur
Pr. Christophe CLARAMUNT	Ecole navale, France	Invité

---

# Remerciements

Tout au long de mon parcours réalisé ces dernières années, j'ai pu bénéficier de soutiens, de conseils et d'encouragements d'un très grand nombre de personnes auxquelles je tiens à exprimer toute ma gratitude.

Je voudrais exprimer ma reconnaissance envers les membres du jury. Pouvoir réunir pour cette occasion des chercheurs d'un tel niveau au sein de disciplines aussi diverses est pour moi un très grand honneur et une marque d'encouragement pour la conduite de recherches interdisciplinaires.

Je voudrais exprimer ma reconnaissance envers Madame Henda BEN GHEZALA professeur à l'école nationale des sciences de l'informatique pour la confiance qu'elle m'a témoignée en acceptant la direction scientifique de mes travaux. Je lui suis reconnaissante de m'avoir fait bénéficier tout au long de ce travail de sa grande compétence, de sa rigueur intellectuelle, de son efficacité et de ses précieux conseils.

Je remercie Monsieur Ali Frihida, maître assistant à l'école nationale des ingénieurs de tunis, pour son aide tout au long de ce projet.

Je tiens à remercier Mr Christophe Claramunt professeur et directeur de l'Institut de Recherche de l'Ecole Navale pour le soutien et les précieux conseils qu'il n'a cessé de me prodiguer tout au long de ce travail.

J'exprime tous mes remerciements à toutes les personnes qui de loin ou de près m'ont aidé à parvenir au terme de ce travail et plus particulièrement Mr Rached Boussemma, Mr Sami Faïez, Mr Saber Naceur, Madame Hajer Baazoui et tous les membres de l'unité de recherche LTSIRS et du laboratoire RIADI.

Je me dois d'exprimer, enfin, ma profonde gratitude aux membres de ma famille qui m'ont accordé leur aide et leur soutien au cours de ces années.

---

## Dédicaces

À ma mère Lilia,  
À mon père Taher,  
À l'âme de mon frère Ikbel,  
À mon mari Slim et mes filles Yasmine et Fatma,  
À tous mes amis,  
À tous ceux que j'aime...  
Et à tous ceux qui veulent partager ma joie...

---

# Table des matières

Table des figures	viii
Liste des tableaux	xi
Introduction	1
<b>I État de l’art</b>	<b>8</b>
<b>I Trajectoires spatio-temporelles d’objets mobiles</b>	<b>11</b>
1 Introduction . . . . .	11
2 Bases de données d’objets mobiles . . . . .	11
2.1 Objets mobiles . . . . .	12
2.2 Exemples d’applications manipulant des objets mobiles . . . . .	15
2.2.1 Applications selon la nature des objets mobiles . . . . .	16
2.2.2 Applications selon la nature de la manipulation des objets . . . . .	16
3 Trajectoires d’objets mobiles . . . . .	17
3.1 Présentation de la notion de trajectoire . . . . .	17
3.2 Moyens d’acquisition des données relatives à la trajectoire . . . . .	18
3.3 Concepts de base des trajectoires spatio-temporelles . . . . .	19
3.3.1 L’espace . . . . .	20
3.3.2 Le temps . . . . .	22
3.3.3 La sémantique . . . . .	24
3.4 Catégories de trajectoires . . . . .	25

3.5	La trajectoire dans le cadre du Time Geography . . . . .	27
4	Conclusion . . . . .	30
<b>II</b>	<b>Modélisation des trajectoires spatio-temporelles</b>	<b>32</b>
1	Introduction . . . . .	32
2	Critères d'évaluation des modèles de représentation des trajectoires . . . . .	33
3	Modèles de trajectoires spatio-temporelles . . . . .	34
3.1	Modèles de trajectoires spatio-temporelles sans apport sémantique . . . . .	34
3.1.1	Modèles par approches formelles . . . . .	35
3.1.2	Modèles par approches non formelles . . . . .	41
3.2	Modèles de trajectoires spatio-temporelles avec apport sémantique . . . . .	43
3.2.1	Modèles par approches formelles . . . . .	43
3.2.2	Modèles par approches non formelles . . . . .	47
4	Synthèse . . . . .	53
5	Conclusion . . . . .	59
<b>II</b>	<b>TST : un type abstrait de données pour les trajectoires spatio-temporelles sémantiquement riches</b>	<b>60</b>
<b>III</b>	<b>Modèle syntaxique du TST</b>	<b>64</b>
1	Introduction . . . . .	64
2	Définition de notre contexte de modélisation . . . . .	64
2.1	Contexte spatial et temporel . . . . .	64
2.1.1	Propriétés spatiales . . . . .	65
2.1.2	Propriétés temporelles . . . . .	65
2.2	Définitions . . . . .	66
2.2.1	Activité . . . . .	66
2.2.2	Déplacement . . . . .	66
2.2.3	Trajectoire spatio-temporelle . . . . .	67
2.3	Contraintes sur la trajectoire spatio-temporelle . . . . .	68
2.3.1	Contraintes spatiales . . . . .	68
2.3.2	Contraintes temporelles . . . . .	68
2.3.3	Contraintes de modélisation . . . . .	69
3	Besoins en analyse des trajectoires . . . . .	70
3.1	Vérification de la cohérence de la base de données . . . . .	70

3.2	Analyse sémantique . . . . .	70
3.3	Analyse spatiale . . . . .	73
3.4	Analyse temporelle . . . . .	74
3.5	Analyse spatio-temporelle . . . . .	75
4	Signature du TST . . . . .	75
4.1	Sorts . . . . .	76
4.2	Opérations . . . . .	78
4.2.1	Opérations de vérification de la cohérence de la base de données	78
4.2.2	Créateurs . . . . .	79
4.2.3	Opérations sémantiques . . . . .	80
4.2.4	Opérations spatiales . . . . .	89
4.2.5	Opérations temporelles . . . . .	94
4.2.6	Opérations spatio-temporelles . . . . .	104
5	Conclusion . . . . .	107
<b>IV Algèbre de définition et de manipulation du TST</b>		<b>108</b>
1	Introduction . . . . .	108
2	Domaine . . . . .	108
3	Sémantique des opérations . . . . .	110
3.1	Opérations de vérification de la cohérence de la base de données . . . . .	110
3.2	Les créateurs . . . . .	111
3.3	Opérations sémantiques . . . . .	112
3.3.1	Opérations de comptage sémantique . . . . .	112
3.3.2	Opérations de positionnement . . . . .	113
3.3.3	Opérations de recherche sémantique . . . . .	115
3.3.4	Opérations de restriction sémantique . . . . .	116
3.3.5	Opérations de projection dans le domaine sémantique . . . . .	119
3.3.6	Opérations de similarité sémantique . . . . .	120
3.3.7	Prédicats sémantiques . . . . .	120
3.4	Opérations spatiales . . . . .	121
3.4.1	Opérations de projection dans le domaine spatial . . . . .	121
3.4.2	Opérations de relations spatiales . . . . .	122
3.4.3	Opérations de voisinage spatial . . . . .	124
3.4.4	Opérations de calcul spatial . . . . .	124
3.4.5	Opérations de recherche spatiale . . . . .	125





<b>B Données Benchmark</b>	<b>189</b>
1 L'enquête Origine-Destination . . . . .	189
2 Pré-traitement des données Benchmark . . . . .	189
2.1 Analogie entre données benchmark initiales et données expérimentales . .	191
2.2 Chargement de la base O-D . . . . .	193
2.3 Traitement des données temporelles . . . . .	193
2.4 Traitement des données spatiales . . . . .	194
2.5 Traitement des données sémantiques . . . . .	194
<b>Bibliographie</b>	<b>196</b>

---

# Table des figures

I.1	Différents types de changements selon les variations de géométrie, topologie et attribut . . . . .	13
I.2	Typologie des changements spatio-temporels . . . . .	14
I.3	Types de changements selon la densité temporelle . . . . .	15
I.4	Exemple du chemin spatial d'un objet mobile . . . . .	18
I.5	Exemple de l'approximation par interpolation linéaire d'une trajectoire dans un espace 3D . . . . .	18
I.6	Relations de topologie spatiales entre objets géométriques . . . . .	21
I.7	Relations de topologies temporelles entre intervalles . . . . .	24
I.8	Exemple d'une trajectoire métaphorique représentant la carrière professionnelle d'un individu . . . . .	25
I.9	Exemple d'une trajectoire géographique naïve représentant le déplacement entre villes . . . . .	26
I.10	Exemple d'une trajectoire spatio-temporelle . . . . .	26
I.11	Mobilité quotidienne et chaîne d'activités et déplacements . . . . .	27
I.12	L'idée de trajectoire spatio-temporelle . . . . .	28
I.13	Trajectoire spatio-temporelle, prisme spatio-temporel et région de trajectoire potentielle . . . . .	30
I.14	Les notions de <i>bundle</i> et de <i>stations</i> dans le time geography . . . . .	30
II.1	Représentation par tranches des valeurs d'un réel mobile et d'un point mobile	37
II.2	Diagramme de classe du projet STAU . . . . .	39
II.3	Types des changements des relations Objet-Réseau . . . . .	40

II.4	Conception orientée objet de véhicules mobiles sur un réseau routier et de leurs trajectoires . . . . .	42
II.5	Partition multi-échelles de l'espace pour l'élaboration d'un patron de mobilité	46
II.6	(a) Trajectoire brute (b) Trajectoire sémantique . . . . .	48
II.7	Modèle conceptuel pour la trajectoire sémantique . . . . .	49
II.8	Modèle conceptuel pour la représentation de données issues de carnets de déplacements . . . . .	50
II.9	Diagramme entité-association d'un modèle de données basé sur les activités .	51
II.10	Modèle de données spatio-temporelles orienté-objet basé sur les activités et déplacements . . . . .	52
II.11	Les composantes de modélisation du type TST . . . . .	62
III.1	Contraintes sur la trajectoire spatio-temporelle d'un objet mobile . . . . .	67
III.2	Un cas de succession temporelle invalide dans la trajectoire spatio-temporelle	69
III.3	Exemple de deux trajectoires sémantiquement similaires . . . . .	72
III.4	Exemple de trajectoires l'une chargée en activités ( $tst_1$ ) et l'autre moins chargée ( $tst_2$ ) et tel que $tst_2$ dure une période de temps plus de longue . . . . .	73
III.5	Exemple d'utilisation des opérations d'ajout et de suppression dans le TST . .	80
III.6	Un exemple de restriction sémantique d'un TST . . . . .	88
III.7	Exemple de projections spatiales du TST . . . . .	96
III.8	Exemple de relations temporelles . . . . .	97
III.9	Exemple d'opérations de restriction temporelle . . . . .	102
IV.1	Exemple de l'opération <code>First_Activity_With_Type</code> . . . . .	119
IV.2	Itinéraire spatial d'un TST . . . . .	121
IV.3	Chemin spatial entre deux activités au sein d'un TST . . . . .	123
IV.4	Exemples d'opérations de recherche spatiale dans un TST . . . . .	129
IV.5	Les différents cas possibles d'Union de deux TST . . . . .	136
V.1	Architecture proposée . . . . .	145
V.2	Structure de données d'un objet de type TST . . . . .	147
V.3	Squelette proposé pour un modèle de la base de données expérimentale . . . .	152
V.4	Modèle final adopté de la base de données expérimentale . . . . .	154
V.5	Zone d'étude : la communauté urbaine du Québec et son réseau routier . . . .	155
V.6	Étapes du processus de pré-traitement des données Benchmark . . . . .	156
V.7	Visualisation des résultats de la requête 1 . . . . .	158

## Table des figures

---

V.8	Visualisation des résultats de la requête 3 . . . . .	161
V.9	Résultats tabulaires de la requête 3 . . . . .	162
V.10	Visualisation des résultats de la requête 4 . . . . .	163
V.11	Résultats tabulaires de la requête 4 . . . . .	164
V.12	Résultats tabulaires de la requête 6 . . . . .	165
V.13	Visualisation des résultats de la requête 7 . . . . .	168
V.14	Résultat de la requête 8 limité à deux enregistrements . . . . .	169
V.15	Visualisation du résultat de la requête 8 . . . . .	170

---

# Liste des tableaux

I.1	Différentes classes de changements spatiaux et/ou temporels selon les variations des attributs, du temps et de la localisation . . . . .	13
II.1	Signature décrivant le système de type abstrait de données pour la représentation des objets mobiles . . . . .	37
II.2	Exemples de signatures de quelques opérations du modèle abstrait du modèle de représentation des objets mobiles par type abstrait de données . . . . .	37
II.3	Tableau comparatif des modèles de représentation de la trajectoire spatio-temporelle . . . . .	58
III.1	Signature des opérations de vérification de la cohérence de la base de données	79
III.2	Signature des opérations de types créateurs de TST . . . . .	80
III.3	Signature des opérations de types comptage sémantique . . . . .	81
III.4	Signature des opérations de positionnement sémantique retournant un évènement	83
III.5	Signature des opérations de positionnement sémantique retournant une liste d'évènements . . . . .	84
III.6	Signature des opérations de positionnement sémantique retournant la position d'un évènement dans le TST . . . . .	84
III.7	Signature des opérations de recherche sémantique retournant un évènement .	86
III.8	Signature des opérations de recherche sémantique retournant une liste d'évènements . . . . .	87
III.9	Signature des opérations de restriction sémantique . . . . .	87
III.10	Signature des opérations de projection dans le domaine sémantique . . . . .	88

## Liste des tableaux

---

III.11	Signature des opérations de similarité sémantique . . . . .	89
III.12	Signature des opérations de types prédicats sémantiques . . . . .	89
III.13	Signature des opérations de projection dans le domaine spatial . . . . .	90
III.14	Signature des opérations de recherche spatiale . . . . .	92
III.15	Signature des opérations de relations spatiales sur un TST . . . . .	94
III.16	Signature des opérations de voisinage spatial . . . . .	95
III.17	Signature des opérations de calcul spatial . . . . .	96
III.18	Signature de l'opération de similarité spatiale . . . . .	97
III.19	Signature des opérations de relations temporelles . . . . .	100
III.20	Signature des opérations de calcul de durées temporelles . . . . .	101
III.21	Signature des opérations de restriction temporelle . . . . .	101
III.22	Signature des opérations de projection temporelle . . . . .	103
III.23	Signature des opérations de recherche temporelle . . . . .	103
III.24	Signature de l'opération de similarité temporelle . . . . .	104
III.25	Signature des opérations spatio-temporelles . . . . .	106
IV.1	Sémantique des opérations de vérification de la cohérence de la base de données	110
IV.2	Sémantique des créateurs . . . . .	111
IV.3	Sémantique des opérations de comptage sémantique . . . . .	112
IV.4	Sémantique des opérations de positionnement sémantique retournant un évènement . . . . .	113
IV.5	Sémantique des opérations de positionnement sémantique retournant une liste d'évènements . . . . .	115
IV.6	Sémantique des opérations de positionnement sémantique retournant la position d'un événement dans le TST . . . . .	115
IV.7	Sémantique des opérations de recherche sémantique retournant une liste d'évènements . . . . .	116
IV.8	Sémantique des opérations de recherche sémantique retournant un événement	118
IV.9	Signature des opérations de restriction sémantique . . . . .	119
IV.10	Sémantique des opérations de projection dans le domaine sémantique . . . . .	120
IV.11	Sémantique de l'opération de test de la similarité sémantique . . . . .	120
IV.12	Sémantique des opérations de type prédicats sémantiques . . . . .	120
IV.13	Sémantique des opérations de projection dans le domaine spatial . . . . .	122
IV.14	Sémantique des opérations de topologie spatiale . . . . .	124
IV.15	Sémantique des opérations de voisinage spatial . . . . .	124

IV.16	Sémantique des opérations de calcul spatial . . . . .	125
IV.17	Sémantique des opérations de recherche spatiale retournant une liste d'événe- ments . . . . .	126
IV.18	Sémantique des opérations de recherche sémantique retournant un événement	128
IV.19	Sémantique de l'opération de similarité spatiale . . . . .	129
IV.20	Sémantique des opération de topologie temporelle . . . . .	131
IV.21	Sémantique des opérations de calcul des durées . . . . .	132
IV.22	Sémantique des opérations de restriction temporelle . . . . .	132
IV.23	Sémantique des opérations de projection dans le domaine temporel . . . . .	133
IV.24	Sémantique des opérations de recherche temporelle . . . . .	134
IV.25	Sémantique des opérations de similarité temporelle . . . . .	134
IV.26	Sémantique des opérations ensemblistes . . . . .	141
IV.27	Sémantique des opérations de voisinage spatio-temporel . . . . .	141
B.1	Données attributaires de la base de données benchmark . . . . .	190
B.2	Données spatiales de la base de données benchmark . . . . .	190
B.3	Données temporelles de la base de données benchmark . . . . .	190
B.4	Données sémantiques de la base de données benchmark . . . . .	191
B.5	Analogie entre données expérimentales finales et données benchmark initiales nécessaires pour la création de la B.D expérimentale . . . . .	192
B.6	Analogie entre données expérimentales finales et données benchmark initiales nécessaires pour la création de la donnée de type TST . . . . .	192

---

# Introduction

## Contexte

Les applications spatio-temporelles, telles que les systèmes de navigation en temps réel et les systèmes d'informations géographiques (SIG), ont en commun un besoin crucial de gestion de données spatiales et de leurs changements à travers le temps. Ces données sont de différents types et de sémantique variable.

Ces dernières années, le développement croissant des technologies de positionnement et des outils de communication sans fils a rendu possible l'acquisition de données sur les positions d'objets en mouvement continu. De telles données, représentent une nouvelle forme d'information, habituellement quantifiée et représentée dans des bases de données d'objets mobiles par des trajectoires spatio-temporelles.

Le besoin d'analyse de trajectoires d'objets mobiles se fait de plus en plus ressentir pour de nombreuses applications, bâties sur la connaissance du mouvement des objets. À titre d'exemple, lorsqu'il s'agit de l'étude de la mobilité de personnes dans un contexte urbain, les données sur les habitudes de déplacements d'un échantillon de personnes serviraient comme données source pour des analyses afin de prévoir des actions de planification urbaines. Lorsqu'il s'agit de l'étude de la mobilité d'animaux, une application de suivi de la migration de horde d'oiseaux serait un autre exemple.

Afin de satisfaire les besoins d'analyse de telles applications, les trajectoires spatio-temporelles doivent idéalement se représenter, dans les bases de données d'objets mobiles, à partir de modèles intégrés de l'espace et du temps. La recherche en bases de données spatio-temporelles en général, et en bases de données d'objets mobiles en particulier a suscité beaucoup d'intérêt ces dernières années. Les travaux proposés se sont surtout focalisés sur la problématique de la



représentation de la continuité du mouvement [61, 71].

L'étude des cadres conceptuels disponibles à cet effet, nous ramène naturellement aux concepts introduits par les principes du *Time Geography* [9, 78]. Le Time Geography est, en effet, un cadre conceptuel théorique accompagné de représentation graphique permettant la description et la compréhension de l'ordonnancement et de la coordination des comportements humains dans l'espace et dans le temps. Dans ce cadre, la notion de trajectoire est conceptualisée à partir de deux types de comportements qui sont les déplacements et les activités reliés entre eux. Miller [35] a été l'un des premiers à utiliser les concepts du Time Geography au sein des SIG, et notamment à partir de la notion de *space-time path* utilisée pour des études d'accessibilité. D'autres s'en inspirent pour modéliser la dynamique des comportements des individus, et capturer la sémantique des applications de transport ou de mobilité en milieu urbain [6, 15, 48].

Cette représentation permet, à notre avis, de compléter les représentations spatio-temporelles classiques des trajectoires avec une connaissance sémantique. Au sein des systèmes de gestion de bases de données spatio-temporelles, les interactions spatiales, temporelles, spatio-temporelles et sémantiques entre ces comportements et ces trajectoires sont rarement exprimées. Ainsi, bien qu'il offre un cadre approprié pour la représentation des différentes dimensions requises pour la trajectoire spatio-temporelle, les concepts du cadre de Time Geography sont rarement formalisés et peu exploités dans les représentations actuelles des bases de données spatio-temporelles.

D'autre part, l'intégration d'une dimension sémantique, offrant une meilleure compréhension des comportements mobiles, a suscité un grand intérêt ces dernières années. Cette sémantique peut s'exprimer sous différentes formes dans la représentation des trajectoires spatio-temporelles. L'existence d'une telle information, permettrait en effet, aux utilisateurs de bases de données d'objets mobiles d'acquérir une meilleure exploitation des données, une meilleure compréhension du mouvement et par conséquent de meilleurs résultats de leurs études.

C'est dans ce contexte que nous avons proposé une modélisation algébrique des trajectoires spatio-temporelles individuelles d'objets mobiles. Cette modélisation offre d'une part, une représentation sémantiquement riche des trajectoires sous la forme d'un type abstrait de données et d'autre part, des fonctions de manipulation à des dimensions spatiales, temporelles et spatio-temporelles ainsi qu'à une dimension sémantique. Le but est d'offrir aux concepteurs de bases de données un type de donnée permettant une représentation unifiée des différentes caractéristiques des trajectoires et un langage de manipulation permettant une manipulation couvrant les différentes dimensions déjà citées.

## Problématique et objectifs

Dans cette thèse, l'intérêt est porté aux problèmes liés à la représentation des trajectoires spatio-temporelles au sein des bases de données d'objets mobiles d'une part et à la représentation des trajectoires dans le cadre particulier de Time Geography, d'autre part.

Ainsi, la thèse s'articule autour d'objectifs émanant d'axes de recherches croisés autour de la modélisation et de la manipulation des trajectoires spatio-temporelles d'objets mobiles. En fait, ces objectifs tentent de répondre aux questions suivantes :

1. Comment intégrer la dimension sémantique dans la représentation des trajectoires spatio-temporelles des objets mobiles ?
2. Comment représenter les trajectoires spatio-temporelles d'objets mobiles afin d'offrir aux utilisateurs de bases de données spatio-temporelles et en particulier aux bases de données d'objets mobiles une représentation :
  - qui d'une part, intègre d'une façon unifiée et cohérente les aspects spatiaux, temporels et surtout sémantique du mouvement,
  - et qui, d'autre part assure un pouvoir et une richesse opérationnelle de la manipulation des trajectoires spatio-temporelles et ceci à différentes dimensions, spatiales temporelles, spatio-temporelles et sémantiques ?

Les premiers travaux proposés pour la représentation des trajectoires avaient comme principal objectif d'assurer l'intégration des composantes spatiales et temporelles à travers une modélisation de la continuité du mouvement [62].

D'autre part, dans les bases de données géographiques, les données spatiales telles que des points ou des lignes, et temporelles telles que des instants de temps, sont mises à la disposition des utilisateurs en tant que entités identifiables et aisément manipulables. Il serait intéressant de faire de même, pour la représentation des trajectoires spatio-temporelles, vu la nature composée de ces données. Certains travaux se sont intéressés à cet aspect [57, 71] et ont proposé des représentations par type abstrait de données des trajectoires d'entités géographiques mobiles. Les modèles proposés sont pour la plupart formellement définis, ce qui assure une rigueur dans la représentation et une certaine généricité qui se traduit par l'indépendance de certains de ces modèles d'un type de SGBD. Néanmoins, nous avons constaté que la plupart de ces travaux négligent la dimension sémantique dans leur représentation et que le peu de travaux qui intègrent la sémantique négligent le côté opérationnel de cette sémantique.

D'un point de vue sémantique, le constat sur l'importance de la prise en considération de cette dimension dans la représentation des trajectoires spatio-temporelles est éminent. De plus

en plus de travaux s'intéressent à cet aspect. Néanmoins, nous avons remarqué que souvent, la sémantique en question est une information dérivée d'informations brutes de la trajectoire, telles que sa direction ou sa vitesse alors qu'une sémantique relative aux besoins applicatifs serait mieux appréciée. En effet, avoir des informations sur le pourquoi et le comment du déplacement serait d'une grande aide pour la compréhension du comportement de mobilité de tout objet mobile.

Nous nous sommes aussi intéressés dans ce travail, au cadre de Time Geography [78]. Dans ce contexte, les trajectoires spatio-temporelles d'objets mobiles est représentée à partir de phénomènes de déplacements et d'activités, liés sémantiquement, étudiés principalement à partir des sciences sociales. Nous estimons que cette représentation offre un cadre intéressant et approprié pour atteindre notre objectif relatif à l'intégration de la sémantique dans les bases de données d'objets mobiles. Peu de travaux se sont intéressés à la représentation des trajectoires dans ce cadre, surtout par des approches formelles. Nous envisageons dans ce travail de recherche d'utiliser et de formaliser la notion de trajectoire spatio-temporelle, telle qu'elle a été conçue dans le cadre du Time Geography, afin de pouvoir associer la caractéristique relative à la sémantique du mouvement dans notre représentation et combler par la même occasion le manque de formalisation des concepts de ce cadre.

Partant de ces constatations, nous nous sommes fixés comme objectifs de proposer une modélisation formelle de la trajectoire spatio-temporelle des données mobiles qui prend en considération l'aspect sémantique du mouvement et de l'intégrer d'une façon unifiée et homogène avec les aspects spatiaux et temporels des trajectoires. Le modèle devra présenter la trajectoire spatio-temporelle comme une entité identifiable, intégrable dans les SGBDs spatiaux au même titre que les informations spatiales et temporelles de base. Il devra aussi la doter de capacités opérationnelles riches à différentes dimensions, spatiales, temporelles, spatio-temporelles et sémantiques. Le cadre de Time Geography devra être utilisé dans cette représentation afin de tirer profit de la sémantique qu'il exprime dans sa vision de la trajectoire spatio-temporelle. Par ailleurs, une validation du modèle proposé, permettant d'illustrer son potentiel de représentation et de manipulation de la donnée trajectoire dans un cadre d'étude particulier et moyennant des données réelles, est nécessaire.

Afin de concrétiser les objectifs évoqués dans cette section, nous proposons l'élaboration d'un type abstrait de données, appelé TST, pour la trajectoire spatio-temporelle. Le type est algébriquement défini et pourra être intégrable dans tout SGBD spatial et mis à la disposition des concepteurs d'applications spatio-temporelles afin de représenter et manipuler, à des dimensions spatiales temporelles, spatio-temporelles et sémantiques, les trajectoires spatio-temporelles d'objets mobiles.

Ce travail de recherche a permis de dégager un ensemble de résultats et de contributions que nous récapitulons dans ce qui suit.

## Contribution

L'état de l'art mené dans le cadre de cette thèse a permis de dresser une étude comparative des principaux modèles de représentation des trajectoires spatio-temporelles existants dans la littérature. Afin de comparer ces approches, nous nous sommes fixés un ensemble de critères comparatifs. Un intérêt particulier a été porté au critère lié à la sémantique des trajectoires.

La classification des modèles étudiés par rapport à ces critères a permis de dégager certains manques dont nous avons tenu compte dans l'élaboration de notre algèbre de représentation et de manipulation de la trajectoire spatio-temporelle d'objets mobiles.

L'algèbre a été proposée pour répondre aux objectifs attendus de cette thèse et élaborés dans la précédente section, à savoir proposer une modélisation par type abstrait de données de la trajectoire spatio-temporelle qui soit :

- formellement définie,
- opérationnellement riche,
- et représentant d'une façon intégrée les aspects spatio-temporels et sémantiques de la trajectoire d'entités mobiles.

Avant d'attaquer la proposition de notre type TST, et pour pouvoir couvrir un maximum d'aspects opérationnels nécessaires lors de la manipulation du type, nous avons commencé par faire une étude sur les besoins d'analyse des trajectoires spatio-temporelles. Nous avons identifié, dans cette étude, différentes dimensions d'analyse et pour chaque dimension, nous avons dégagé différentes catégories de besoins de manipulations de la trajectoire spatio-temporelle.

Sur la base de cette étude, nous avons proposé notre type TST en deux étapes :

1. d'abord, une spécification formelle du TST est donnée. Cette spécification représente la *signature* ou le *modèle syntaxique* du type proposé.
2. ensuite l'algèbre liée à cette signature est spécifiée. L'algèbre proposée représente le *modèle sémantique* de notre type. Autrement dit, elle définit les données constituant le TST et exprime le comportement de ses opérations de manipulations à l'aide de formalisations mathématiques offrant une rigueur et une indépendance dans la représentation d'un type particulier de SGBD.

Des algorithmes associés aux opérations de manipulations du TST ont été proposés. Le

type TST a été implémenté au sein du SGBD open source PostgreSQL et a fait l'objet d'une expérimentation dans le cadre d'une étude de la mobilité au centre ville et avec des données benchmark.

L'élaboration de tous ces résultats fera l'objet des différents chapitres de ce mémoire.

## Organisation du mémoire

Après une introduction générale de la thèse, ce document est organisé en cinq principaux chapitres organisés en deux parties.

La première partie présente l'état de l'art sur les trajectoires spatio-temporelles et leurs modélisations. Elle est composée de deux chapitres :

- **Le premier chapitre** décrit l'état de l'art sur la notion clé de notre mémoire, à savoir la trajectoire spatio-temporelle d'objets mobiles. Les caractéristiques de la trajectoire spatio-temporelle et ses différentes catégories sont également dégagées au sein de ce chapitre.
- **Le deuxième chapitre** offre un tour d'horizon sur un ensemble de modèles de représentation de la trajectoire d'entités mobiles. Une étude comparative des ces modèles est proposée et discutée replaçant, ainsi, notre modèle dans son contexte.

La deuxième partie est consacrée aux contributions et expose les propositions en termes de modélisation algébrique des trajectoires spatio-temporelles. Le modèle algébrique proposé est présenté dans cette partie en deux étapes : d'abord le modèle syntaxique, ensuite l'algèbre exprimant la sémantique des composantes du modèle syntaxique. Cette partie expose aussi la validation expérimentale du modèle proposé. Elle est composée de trois chapitres :

- **Le troisième chapitre** présente le modèle syntaxique du type proposé. Ce modèle représente la signature du type sous la forme d'un ensemble de types et d'un ensemble d'opérations qui utilisent ces types et offrent une interface pour la manipulation du TST. Pour ce faire, et afin de pouvoir proposer un aspect opérationnel riche qui recouvre les principaux aspects de la trajectoire, une étude sur les besoins d'analyse de la trajectoire spatio-temporelle d'objets mobiles est présentée.
- **Le quatrième chapitre** décrit le modèle sémantique du type proposé. Autrement dit, il propose l'algèbre liée à la signature, précédemment décrite, en précisant les domaines de valeurs des types et la sémantique mathématiquement formulée de ses opérations.
- **Le cinquième chapitre** est consacré à la présentation de la mise en oeuvre de notre modèle au sein du SGBD spatial PostgreSQL/PostGIS. Une validation expérimentale du modèle développé est aussi présentée. Le but de cette validation est de montrer le potentiel

et les possibilités de notre modèle de représentation et de manipulation de trajectoires dans un cadre d'étude réel et avec des données réelles. Ce chapitre présente également la formulation d'un ensemble de requêtes exprimées en utilisant le langage de manipulation proposé et en expose les résultats. Une section consacrée au positionnement du modèle par rapport à l'existant est aussi proposée.

Le mémoire est clôturé par une conclusion qui annonce des perspectives prometteuses de prolongement de ce travail.

Première partie

État de l'art





---

Les applications spatio-temporelles gèrent des données qui sont caractérisées par des changements dans l'espace et dans le temps. Lorsqu'il s'agit de changements de localisations dans le temps, qu'ils soient discrets ou continus, on parle alors d'objets mobiles. Avec l'avènement des technologies d'acquisition de données en temps réel, la représentation de ces données et de leurs trajectoires est devenue une nécessité. De ce fait, de plus en plus de travaux se sont intéressés à cet axe de recherche.

Nous présentons dans cette partie l'état de l'art sur les trajectoires spatio-temporelles et leur modélisation. En effet, le premier chapitre introduit la notion de trajectoire spatio-temporelle d'objets mobiles à partir, tout d'abord, d'une présentation des bases de données des objets mobiles, bases de données où les trajectoires sont la donnée essentiellement manipulée. Par la suite, le chapitre, identifie les concepts de base des trajectoires spatio-temporelle et ses différentes catégories. Un intérêt particulier est porté à la notion de trajectoire dans le cadre du Time Geography [78].

Le second chapitre, présente une panoplie des travaux de modélisation des trajectoires spatio-temporelles d'objets mobiles. Le chapitre donne une description de chaque modèle sur la base de critères préalablement définis. Une synthèse vient conclure ce chapitre par une étude comparative de ces mêmes modèles.

---

---

# Chapitre I

---

## Trajectoires spatio-temporelles d'objets mobiles

### 1 Introduction

Les applications spatio-temporelles, en particulier les applications mobiles ont un besoin crucial de représentation et de gestion de données représentant les dynamiques spatio-temporelles des entités du monde réel. Afin de mieux aborder cette problématique, nous nous intéressons dans ce chapitre, en premier lieu, aux bases de données d'objets mobiles et à ce qu'ils gèrent comme données. Nous identifions, par la suite, un concept important lié à la représentation des changements dans les bases de données d'objets mobiles, qui est la trajectoire spatio-temporelle. Nous présentons et introduisons, enfin, quelques concepts de base des trajectoires spatio-temporelles d'objets mobiles, notamment les trajectoires dans le cadre spécifique du Time Geography.

### 2 Bases de données d'objets mobiles

En raison du développement considérable des technologies d'acquisition des données mobiles, la gestion de données se déplaçant dans l'espace et dans le temps est rendu possible. En effet, des outils tels que les systèmes de géolocalisation ou les outils de communication sans fil permettent la transmission et la communication d'un grand volume d'informations sur des objets mobiles. Ces données sont représentée dans, ce qui est appelé, des bases de données d'objets mobiles. Les problématiques liées à ce type de bases de données ont suscitées ces dernières années un grand intérêt. Afin de mieux aborder ces problématiques par la suite, nous présentons dans ce qui suit les caractéristiques des données gérées dans les bases de données mobiles et nous donnons quelques exemples d'applications.

### 2.1 Objets mobiles

Les objets mobiles sont représentés dans les bases de données par des données spatio-temporelles. Le terme objet, que nous utilisons ici, ne fait pas référence à la notion utilisée dans le raisonnement orienté objet. Nous utiliserons aussi, plus tard, le terme entités mobiles pour désigner des objets mobiles.

Les données spatio-temporelles sont des données caractérisant des entités géographiques subissant des changements au cours du temps. À titre d'exemples, une parcelle de terre, dont les attributs changent au cours des années, et un véhicule, dont les localisations spatiales changent au cours de la journée, sont considérés comme des données spatio-temporelles.

D'une façon générale, un changement survient lorsqu'on constate une *évolution* significative des caractéristiques d'une entité au cours du temps. Langran [40] souligne l'importance d'une définition précise de la notion de changement pour établir un lien entre le temps et l'espace. De ce fait, plusieurs travaux se sont intéressés à la caractérisation et à la classification du changement subi par les phénomènes géographiques. Nous allons, dans ce qui suit, passer en revue une panoplie de ces travaux.

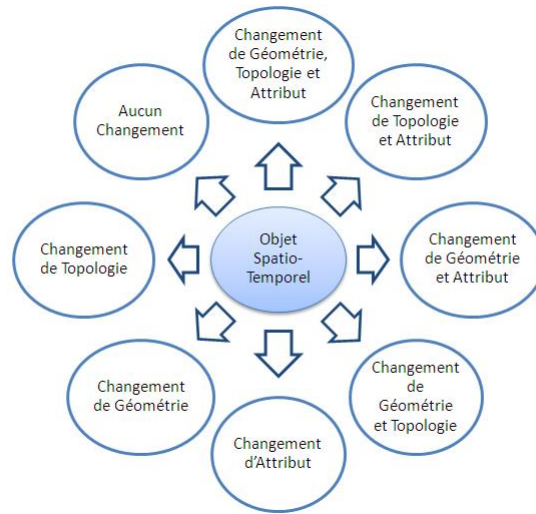
Dans un système d'informations géographiques, une information géographique s'exprime souvent sous la forme :

- d'une information *descriptive* sous forme alphanumérique,
- d'une information *géométrique* de position et de forme dans l'espace et sur la terre exprimée sous forme numérique et graphique,
- et d'une information *topologique* définissant les relations de voisinage, d'inclusion et d'intersection entre objets géométriques.

Abler & al [68] classent les changements subis par un objet géographique selon l'une de ces caractéristiques, à savoir la forme géométrique. Cette classification a contribué à comprendre le changement des entités dans l'espace. Néanmoins, elle ne prenait pas en considération la nature temporelle et dynamique des phénomènes. D'autres [8], différencient huit scénarios de changements selon les variations de la forme, la topologie et les attributs d'un objet géographique dans le temps (voir Figure I.1).

D'autre part, Sinton [18] met en avant l'importance de l'information temporelle lors de la modélisation de la dynamique des phénomènes naturels et des changements qu'ils subissent. Ainsi, il définit les entités géographiques par leurs trois composantes qui sont les *attributs*, la *localisation* et le *temps*. La mesure du changement revient alors à fixer une composante, contrôler l'autre et mesurer la dernière.

Sur la base de cette approche, Yuan [50] identifie six classes de changements spatiaux et/ou



**Figure I.1** – Différents types de changements selon les variations de géométrie, topologie et attribut

temporels (voir Table I.1). La première classe est relative aux modifications des attributs dans le temps ; l'espace quant à lui reste invariant. La seconde décrit la distribution statique des phénomènes géographiques comme par exemple la topographie. Les quatre dernières classes reflètent le changement de géométrie ou de topologie des propriétés temporelles et/ou spatiale. Les changements spatiaux présentés sont de type *statique* ou *transitionnel*. Les changements temporels décrivent une *mutation* ou un *mouvement*.

Classe de changement	Fixé	Contrôlé	Mesuré
I. Fixé	Localisation	Attributs	Temps
II. Catégorie	Temps	Attributs	Localisation
III. Statique	Temps	Localisations	Attributs
IV. Transitionnel	Attributs	Localisations	Temps
V. Mutation	Localisations	Temps	Attributs
VI. Mouvement	Attributs	Temps	Localisations

**Tableau I.1** – Différentes classes de changements spatiaux et/ou temporels selon les variations des attributs, du temps et de la localisation

Dans une description du processus spatio-temporel, d'autres études [2] avancent trois concepts qui, selon les auteurs, semblent suffisants pour caractériser le changement spatial lorsqu'il affecte des entités. Ces concepts sont la *vie* d'un objet, le *mouvement* et la *généalogie* d'un objet. Le terme "mouvement" regroupe ici tous les changements de forme que l'objet observé a subi mais qui n'affectent pas son identité. Dans cette acception, la notion de mouvement incorpore un grand nombre de transformations élémentaires susceptibles de contribuer à ces changements :

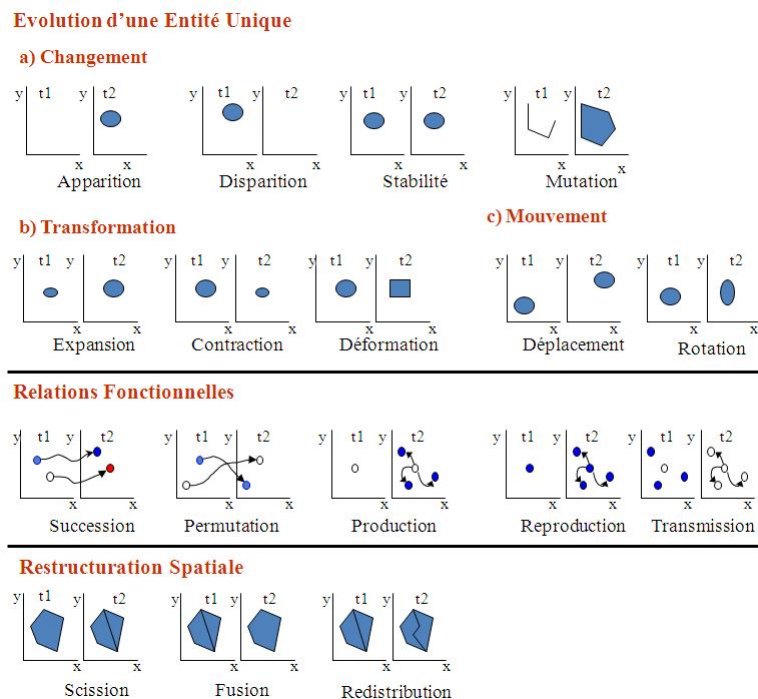
## Chapitre I. Trajectoires spatio-temporelles d'objets mobiles

mouvement de l'objet lui-même, à géométrie constante (translations, rotations), changement de niveau d'organisation spatiale (le point devient ligne, ou polygone, le nombre de points se modifie), changement de frontière.

Dans une approche plus spécifique, Thériault & al [49] proposent une typologie où les processus élémentaires représentant le changement forment trois classes fondamentales (voir Figure I.2) :

- ceux qui décrivent l'évolution d'une entité indépendante en termes de :
  - *changements substantiels* : apparition, disparition, stabilité et mutation ;
  - de *transformations* : expansion, contraction, déformation ;
  - et de *mouvements* : déplacement et rotation.
- ceux qui impliquent des *relations fonctionnelles* entre plusieurs entités interdépendantes, comme la *succession*, la *permutation* ou la *production*,
- et ceux qui manifestent des *restructurations spatiales*, comme la *scission* ou la *fusion*.

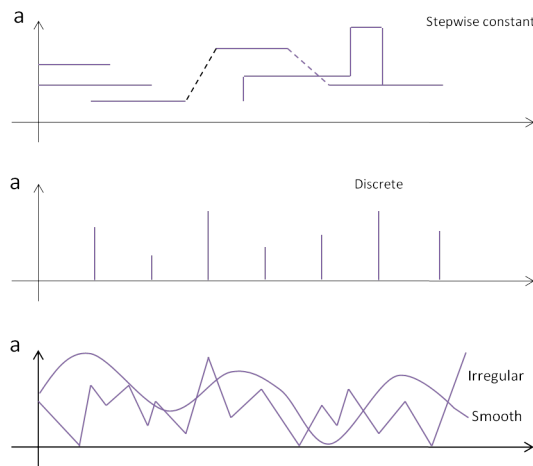
Les auteurs précisent ici que la description de l'évolution est nécessaire pour toutes les applications de systèmes d'informations géo-historiques.



**Figure I.2** – Typologie des changements spatio-temporels

Dans [54], une classification du changement sur la base de la densité temporelle est présentée (voir Figure I.3). Les auteurs s'appuient sur les deux modèles de représentations du temps :

*discret* et *continu* (voir section 3.3.2) pour caractériser la nature des événements. Dans la représentation discrète, les entités ayant été exposées à des événements soudains subissent des changements de type *stepwise*. Les événements continus, quant à eux, sont divisés en deux sous-catégories : *uniformes* et *irréguliers* (i.e le mouvement de taxis dans les villes). Un troisième type, appelé *discret*, expose les valeurs discrètes collectées à des intervalles de temps réguliers ou périodiques.



**Figure I.3** – Types de changements selon la densité temporelle

Bien que dans la littérature il existe plusieurs classifications du changement, nous remarquons qu'il en ressort tout de même que la conceptualisation du changement ramène souvent à la définition du mouvement, qu'il soit discret ou continu.

Les objets subissant un changement de type mouvement sont appelés objets mobiles. Les bases de données qui gèrent le mouvement sont des bases de données d'objets mobiles. Le mouvement d'objets mobiles est quantifié par des *trajectoires* qui tracent et représentent les localisations des objets à travers l'espace et le temps. Les bases de données d'objets mobiles sont alors des bases qui stockent essentiellement des trajectoires.

Nous présenterons, plus en détail, dans la section 3 la notion de trajectoire spatio-temporelle, notion clé dans notre travail.

## 2.2 Exemples d'applications manipulant des objets mobiles

Les domaines d'application s'intéressant à la gestion de la mobilité des objets sont multiples. Nous distinguons différentes classifications de ce type d'application selon la nature des objets qu'ils gèrent [67] ou la nature de la manipulation qu'ils effectuent sur ces données [10].

### 2.2.1 Applications selon la nature des objets mobiles

Selon que l'objet mobile soit un *animal*, un *véhicule*, une *personne* ou une entité géographique, Laube & al [67] identifie différentes formes d'applications :

- Lorsqu'il s'agit de l'étude de la mobilité d'animaux ou de différentes forme de vie (i.e des plantes), des applications relevant du domaine de la biologie ou de l'écologie peuvent être concernées. Il peut s'agir aussi d'applications de suivi de l'immigration de hordes d'animaux.
- Les applications de gestion de véhicules en mouvement peuvent être retrouvées, par exemple, dans les domaines urbains, maritimes ou militaires. Dans de tels systèmes, le suivi des itinéraires d'une flotte de véhicules ou de navires et l'analyse de ces données est d'une grande importance pour les décideurs.
- On retrouve les applications manipulant des données représentant des individus en mouvement, par exemple, dans le domaine urbain. Dans ce cas, les comportements et les habitudes de déplacements d'un échantillon de personnes sont analysés afin de prévoir des actions de planification urbaines.
- Le domaine géographique et ses différentes disciplines, comme la géomorphologie, s'intéressent aux changements subis par les entités géographiques, telles que des parcelles de terre, à travers le temps. Ces changements sont souvent de nature discrète car peu fréquents dans le temps.

### 2.2.2 Applications selon la nature de la manipulation des objets

Les applications qui gèrent des objets mobiles peuvent être classées différemment, selon leur finalité de manipulation. En effet, Du Mouza [10] distingue deux classes d'applications :

- Les applications qui s'intéressent à l'analyse à posteriori du mouvement d'entités mobiles. Dans ce cas, l'historique du mouvement, décrit par les trajectoires spatio-temporelles des objets mobiles, est l'information stockée et manipulée. Une requête spatio-temporelle envisageable serait alors : "Donner tous les objets dont la trajectoire passe dans la région  $R$  entre les instants de temps  $t1$  et  $t2$ ".
- Les applications orientée vers le suivi des objets en temps réel. L'intérêt est porté, dans ce genre d'application, à la position courante de l'objet mobiles. Les manipulations ne sont plus de type "analyse historique du mouvement" mais sont plutôt orientées vers la prédiction de situations dans un avenir proche. Une requête possible de ce genre d'applications serait : "Trouver tous les avions qui vont entrer dans la région  $R$  dans moins de 10 minutes."

## 3 Trajectoires d'objets mobiles

L'analyse des données relatives aux trajectoires devient primordiale pour un grand nombre d'applications dont l'objectif est de gérer des phénomènes complexes impliquant des objets mobiles [76]. Afin d'assurer une manipulation efficace de cette nouvelle forme de données, qui est la trajectoire, et afin de cerner les besoins en terme de modélisation, il est nécessaire de présenter cette notion et d'identifier ses concepts de base.

Dans cette section, nous définissons et présentons les particularités des trajectoires d'entités mobiles. Nous nous intéressons en particulier à la représentation des trajectoires dans le cadre particulier de Time Geography.

### 3.1 Présentation de la notion de trajectoire

Une trajectoire peut être perçue comme un enregistrement ou une trace historique de l'évolution des positions d'une entité qui se déplace dans un certain espace, durant une durée de temps.

Dans les bases de données d'objets mobiles, l'entité mobile est généralement géométriquement représentée comme un point (par exemple une personne, un animal, une voiture, etc.). Mais l'entité mobile peut avoir une surface (par exemple un cercle, un rectangle, etc.) ou un volume géométrique (par exemple un nuage, la pollution de l'air, une avalanche, etc.). Dans ce cas, en plus des changements de position, les changements de forme peuvent contribuer à la définition de la trajectoire. Dans notre travail, nous nous restreignons au premier cas et nous étudions les trajectoires d'objets ponctuels mobiles.

L'"historisation" du mouvement peut passer par la représentation d'une séquence ordonnée des différentes localisations atteintes par l'objet en mouvement ; c'est le chemin spatial (voir Figure I.4). Sauf qu'un chemin n'est jamais créé instantanément mais nécessite toujours la consommation d'une certaine quantité de temps. Le temps est, donc, un aspect inséparable de la trajectoire et les caractéristiques temporelles sont des paramètres essentiels dans la définition des trajectoires.

La trajectoire d'un objet mobile peut alors être représentée de deux manières :

- Dans la première solution, le modèle discret du temps est adopté. Dans ce cas, seule une séquence finie de  $\langle \text{temps}, \text{position} \rangle$  est stockée. Cette séquence indique les positions atteintes au cours du mouvement et les temps d'observation de ces positions. Bien entendu, cette solution ne permet pas d'associer à tout moment temporel une position spatiale mais ceci peut être suffisant pour certains types d'applications. En cas de besoin et afin d'ob-



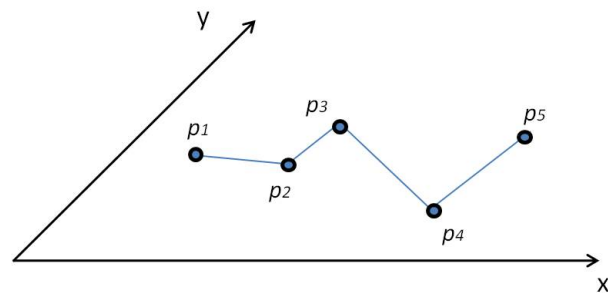


Figure I.4 – Exemple du chemin spatial d'un objet mobile

tenir la totalité de la trajectoire, une approximation de la trajectoire peut être effectuée, par exemple, par interpolation linéaire (voir Figure I.5).

- Dans la seconde solution, le modèle de représentation du temps est plutôt temporel. La trajectoire est alors une fonction,

$$f: \text{temps} \rightarrow \text{espace}$$

qui fait correspondre à tout instant du temps une position de l'espace.

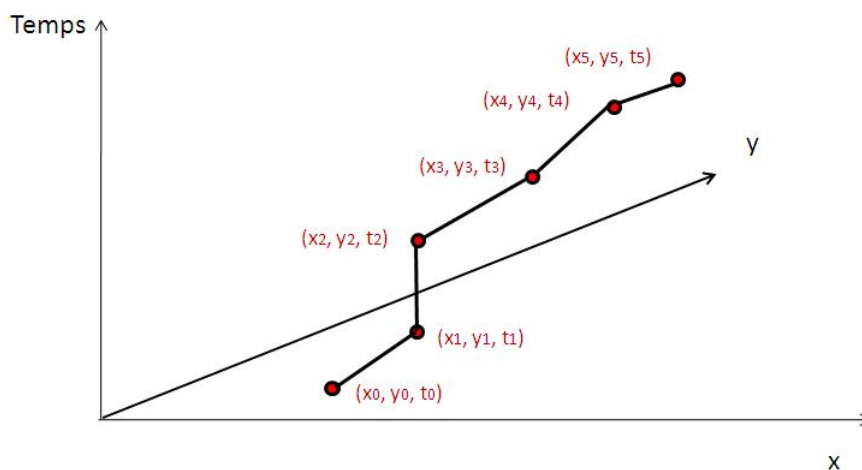


Figure I.5 – Exemple de l'approximation par interpolation linéaire d'une trajectoire dans un espace 3D

### 3.2 Moyens d'acquisition des données relatives à la trajectoire

Les données *primitives* ou *brutes* composant l'essentiel des données de la trajectoire, à savoir les positions spatiales et leur temps d'observation, peuvent être obtenues par divers moyens de collecte de données mobiles :

- Les capteurs GPS (Global Positioning System), les téléphones mobiles et les différents capteurs mobiles permettent, en effet, de produire, stocker et communiquer des traces sur la mobilité d'un objet. À titre d'exemple, avec ce type d'outils on pourrait avoir la trajectoire de voitures équipées de GPS ou d'animaux équipés de transmetteurs dont le signal est capturé par satellite. De telles trajectoires peuvent être exploitées par des applications de transport, des applications militaires ou des applications d'étude du comportement d'oiseaux migrateurs.
- Les personnes peuvent être questionnés sur leur mobilité : quand et où ils se déplacent. Ce sont les enquêtes Origine/Destination qui représentent un moyen direct pour avoir l'itinéraire d'individus [25]. Le résultat est un journal d'activité/déplacement indiquant les instants de déplacements d'un endroit à un autre et les instants d'exercice de certaines activités. Bien sûr, la localisation d'individus peut aussi, de nos jours, être possible grâce à la combinaison des divers outils "*location aware*" cités précédemment (e.g., capteurs GPS, téléphonie mobile, etc.).

D'autre part, selon [52], les moyens d'observation des mouvements peuvent être classés selon le mode d'enregistrement des données. En effet, on retrouve des moyens :

- d'enregistrements basés sur le temps : les positions des entités sont enregistrées à des instants temporels régulièrement espacés, par exemple à chaque 5 minutes ;
- d'enregistrements basés sur le changement : il y a enregistrement lorsque la position d'une entité change par rapport à sa position précédente ;
- d'enregistrements basés sur la localisation : un enregistrement est effectué quand l'entité se rapproche de certaines localisations bien spécifiques, à titre d'exemple des endroits où des détecteurs sont installés ;
- d'enregistrements basés sur l'événement : les positions spatiales et le temps sont enregistrés lorsque certains événements se produisent, à titre d'exemple lorsque l'entité mobile exerce des activités comme de téléphoner avec un téléphone portable ;
- ou par diverses combinaisons de ces approches.

### 3.3 Concepts de base des trajectoires spatio-temporelles

Dans notre présentation des trajectoires spatio-temporelles, dans ce qui a précédé, certaines caractéristiques ont été naturellement mises en avant, à savoir les caractéristiques spatiales et temporelles des trajectoires. D'autre part, la majorité des recherches sur les données mobiles reconnaissent la nécessité de prendre en considération non seulement ces caractéristiques mais aussi la structure et les propriétés propres à l'espace et au temps (dans lesquels a lieu le

mouvement) eux-mêmes. D'autre part, de plus en plus de travaux récents [13] estiment que les deux composantes, spatiale et temporelle, sont insuffisantes et qu'une trajectoire doit, en plus, comporter une composante sémantique.

Nous identifions et nous présentons dans cette section les principales caractéristiques des trajectoires spatio-temporelles, ou du moins celles qui peuvent placer dans une perspective générale les recherches sur la modélisation dans les bases de données d'objets mobiles.

### 3.3.1 L'espace

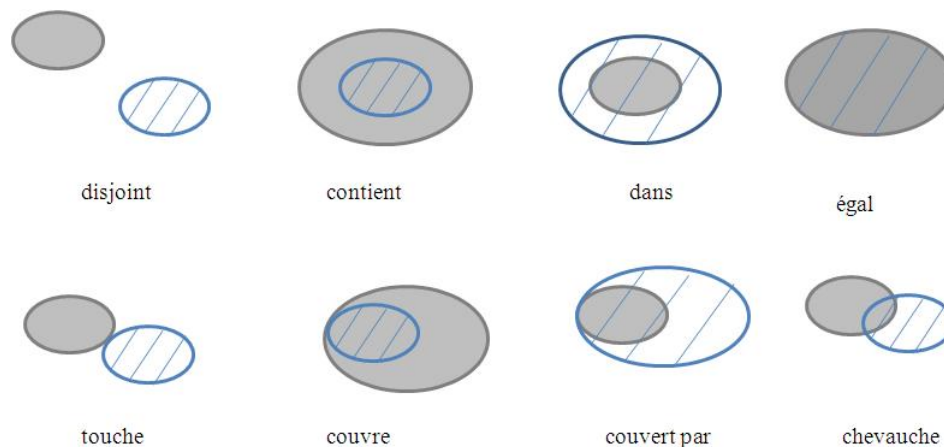
Selon le Grand Robert, l'espace est un "Lieu, plus ou moins bien délimité, où l'on peut se situer quelque chose". Notre intérêt porte ici sur l'espace géographique. L'espace est une caractéristique essentielle des trajectoires spatio-temporelles, vu qu'un mouvement ne peut avoir lieu que dans un espace bien déterminé. Nous exposerons donc dans cette partie les caractéristiques spatiales des trajectoires spatio-temporelles. D'autre part, la définition de l'espace géographique ne peut être pertinente que si l'on expose les propriétés qui lui sont propres.

#### Propriétés de l'espace

- *Référentiel spatial* : Un objet mobile possède différentes positions dans l'espace où a lieu le mouvement. Pour pouvoir distinguer ces positions et mesurer les distances entre elles, il faut se munir d'un *système de référence*. Une trajectoire spatio-temporelle est habituellement décrite dans un référentiel spatial *basé sur une localisation* ou à partir d'une représentation *relative au sein d'un réseau*. Dans le premier cas, les données spatiales de la trajectoire sont représentées par des coordonnées et/ou des primitives géométriques (e.g., série de points, polygones). Dans le deuxième cas, une trajectoire est représentée à partir de la structure logique du réseau de transport et de ses contraintes, à savoir à l'aide d'arcs et de noeuds.
- *Espace continu/discret* : L'espace physique est de nature continue, c'est-à-dire qu'il est composé d'un nombre infini de localisations et qu'entre deux localisations il existe toujours d'autres localisations. Néanmoins, l'espace peut être considéré comme discret en ne considérant qu'un ensemble fini de localisations.
- *Dimension* : Selon les besoins applicatifs, l'espace peut être uni-dimensionnel, bi-dimensionnel ou tri-dimensionnel. Dans le premier cas une seule coordonnée est suffisante pour définir les positions des objets. À titre d'exemple, le cas spécifique où le mouvement est étudié le long d'une route bien déterminée et où les positions sont définies par la mesure des distances à partir du début de la route. Dans un espace bi-dimensionnel chaque position

est définie par une paire de coordonnées et dans un espace tri-dimensionnel les positions sont définies par un triplet de coordonnées.

- *Echelle spatiale* : L'échelle spatiale, ou échelle géographique ou encore granularité spatiale, consiste en le rapport de taille entre deux réalités géographiques. Elle détermine le niveau du détail spatial considéré. Le choix de l'échelle spatiale lors d'une étude dépend de la nature et des objectifs de l'application.
- *Structure* : Selon le besoin, l'espace peut avoir une *structure* particulière, à titre d'exemple il peut être divisé hiérarchiquement en zones d'intérêts, déterminées par un thématicien en fonction des objectifs de son étude. À titre d'exemple, un pays peut être divisé en gouvernorats, les gouvernorats en délégations et les délégations en municipalités.
- *Relations spatiales* : Les relations spatiales entre les objets du monde réel, objets représentés par des entités géométriques (points, lignes et régions), peuvent être classées en trois catégories principales : les relations métriques (e.i *proche de*, *à telle distance de*, etc.), les relations d'orientation (e.i *à gauche de*, *à droite de*, etc.) et les relations topologiques (e.i *est inclus dans*, etc.) [83]. En effet et à titre d'exemple, Egenhofer [43] a développé dans les années 90 le modèle des neuf intersections qui permet de représenter les relations entre deux régions (voir Figure I.6), deux lignes [41] et entre une ligne et une région [42]. Comme pour des entités spatiales, il serait intéressant que des trajectoires différentes d'un même objet mobile ou des trajectoires de différents objets mobiles puissent aussi être comparées par rapport à leurs situations spatiales.



**Figure I.6** – Relations de topologie spatiales entre objets géométriques

**Les caractéristiques spatiales des trajectoires** : Dans l'espace d'étude, la trajectoire d'un objet en mouvement peut posséder un nombre de caractéristiques liées à l'espace. Nous

en recensons quelques-unes :

- la *position* de l'objet mobile dans l'espace ;
- la *vitesse*, la *direction* et l'*accélération* du mouvement ;
- la *distance* totale ou partielle accumulée par le mouvement.

### 3.3.2 Le temps

Le mouvement implique une consommation du temps. Le temps est donc une propriété primordiale des trajectoires spatio-temporelles. La définition du temps a toujours comporté des difficultés ontologiques bien connues des philosophes du 19ème siècle et pour lesquelles les sciences modernes n'a toujours pas de solution universelle. Toujours associé à l'espace, le concept de temps était tantôt perçu comme un simple "conteneur " des événements du monde réel (vue absolue) tantôt comme une abstraction permettant de représenter les relations entre les objets et les processus (vue relative) [24]. De nos jours, la coexistence de plusieurs conceptions complémentaires ou concurrentes du temps complique son informatisation. En effet, cette dernière requiert une formalisation mathématique des composantes temporelles afin de leur appliquer les opérations numériques, logiques, combinatoires et ensemblistes. Face à cette difficulté de représentation, il est primordial de comprendre les différents aspects du temps, particulièrement si on essaie de modéliser les changements spatio-temporels et plus particulièrement les trajectoires spatio-temporelles.

#### Propriétés du temps

- *Nature du temps* : Il existe deux conceptions majeures de la nature du temps. La conception *newtonienne* ou approche *absolue* considère le temps comme un flux mesurable et quantifiable (heures et dates) sur une échelle d'intervalles. Elle privilégie une vue chronologique basée sur la mesure du temps, avec une granularité adaptée aux domaines (heures, jours, année, etc.). D'autre part, la conception *Leibnizienne* ou approche *relative* appréhende le temps par la succession des événements, afin de déterminer des séquences exprimées sur une échelle ordinale (avant, simultanément, après). Cette approche favorise une vue historique basée sur la temporalité perçue à travers l'ordre des événements. Chacune de ces formes offrent un puissant pouvoir sémantique.
- *Entités primitives temporelles* : La définition d'entités primitives est l'un des aspects les plus fondamentaux et les plus discutés de l'ontologie du temps [24]. Selon la nature perçue du temps, la littérature distingue les primitives suivantes :
  - des **points** de temps (*instants*) ;

- des **intervalles** de temps (*périodes*);
- des **occurrences** dans le temps (*événements*).

En effet, dans la conception absolue du temps, qui est une approche quantitative, le temps peut être divisé en instants ou en périodes. Les instants sont une représentation discrète et servent à identifier un moment ou une date particulière. Les périodes correspondent à une perception continue et servent à exprimer des durées. Par ailleurs, la conception relative, qui est une approche ordinale, représente le temps comme un ensemble d'événements ordonnés.

- *La topologie du temps* : Étant donné le caractère dynamique du temps, il est naturel de définir des trajectoires dans le temps pour lesquelles deux entités temporelles sont reliées par une *relation d'ordre*. Cet ordre peut être *total* ou *partiel*. Dans le premier cas, chaque entité temporelle possède au plus un prédécesseur et un successeur; cette propriété peut être vérifiée par une représentation *linéaire* soit par une représentation *circulaire*, dite aussi *périodique*, du temps. La linéarité du temps sert à représenter le progrès et l'évolution dans le temps, tandis que la périodicité exprime la constance et la continuité. Dans le cas où la relation d'ordre utilisée est partielle, chaque entité temporelle peut posséder plusieurs prédécesseurs et successeurs. On parle de *temps ramifié* (*branching time*).
- *Relations/opérations temporelles* : Avec une description du temps sous forme de points, une simple relation de précédence est requise. La description sous forme d'intervalles nécessite un ensemble d'opérations, décrites par Allen [23] (voir Figure I.7).
- *Structure ou densité du temps* : Cette caractéristique distingue entre le temps *discret* et le temps *continu*. Dans la structure discrète, le temps est mesuré en certains points temporels ou intervalles et la variation est discontinue entre ces points; par conséquent les points dans ce temps sont isomorphes aux nombres naturels. Dans le cas du temps continu, la structure du temps est isomorphe aux nombres réels et ne contient donc pas d'interruptions.
- *Granularité temporelle* : Nous pouvons définir la granularité temporelle comme le degré de résolution temporelle. Dans [30], la granularité est spécifiée par un point ancré sur l'axe de temps et une longueur de division. Le point ancré dénote où la division commence tandis que la longueur de division dénote la taille de chaque granule. Des applications différentes exigent des niveaux différents de granularité.
- *Temps Valide ou Temps de transaction* : Le *temps valide* correspond au temps d'occurrence d'un fait dans la réalité. Le *temps de transaction* est le temps auquel les données concernant le fait ont été conservées dans la base de données [70].

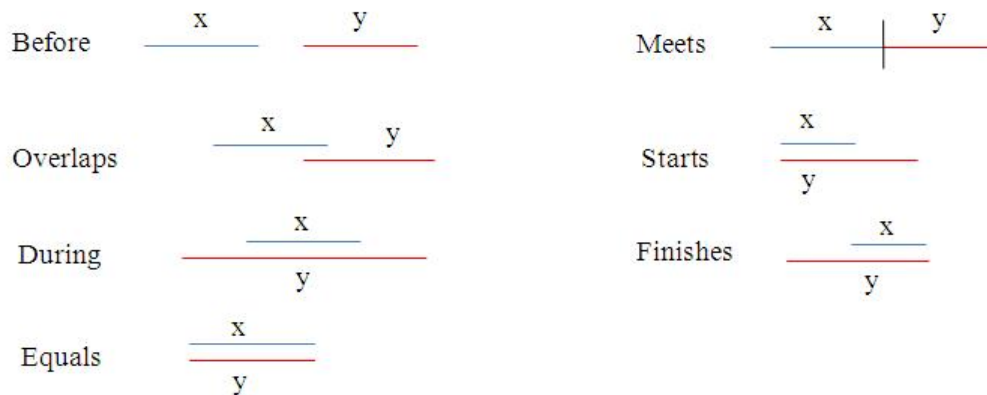


Figure I.7 – Relations de topologies temporelles entre intervalles

**Les caractéristiques temporelles des trajectoires** Étant donné un temps d'étude, la trajectoire d'un objet en mouvement peut posséder un nombre de caractéristiques liées au temps. Nous en recensons quelques-unes :

- le *temps du mouvement*, qui est par exemple la position de ce moment sur l'échelle temporelle ;
- la *durée temporelle* totale ou partielle du mouvement ;
- la *représentation du temps* : la question qui se pose ici est comment et à quel niveau les caractéristiques temporelles des trajectoires sont-elles intégrées dans les modèles de trajectoire spatio-temporelle. Différentes approches peuvent être adoptée comme l'horodatage (time-stamping) temporels des objets géographiques ou la fusion de l'espace et du temps dans des objets spatio-temporels unifiés [85]. La première solution est moins coûteuse en terme de stockage mémoire mais plus limitée par rapport au pouvoir d'expression des propriétés temporelles de l'objet.

### 3.3.3 La sémantique

Une trajectoire peut exprimer explicitement différents types de sémantique. Cette sémantique peut être dérivée des informations spatiales d'une trajectoire, comme la direction ou la vitesse. Elle peut aussi concerner des localisations ayant une signification particulière, concerner des motifs ou modes de déplacements ou concerner des informations liées aux besoins des applications (e.g., activités). Prendre en considération ces différents aspects sémantiques lors de la représentation des trajectoires permettrait d'enrichir les modèles de trajectoires spatio-temporelles et d'améliorer le potentiel d'analyse de cette donnée.

### 3.4 Catégories de trajectoires

Selon la nature de l'espace dans lequel la mobilité est exercée, Spaccapietra & al [76] distinguent trois catégories de trajectoires que voici :

#### Les trajectoires métaphoriques

Il s'agit ici d'un usage métaphorique du terme trajectoire. Selon les auteurs, une trajectoire métaphorique décrit une évolution qui n'est pas liée à un mouvement physique mais plutôt à un déplacement dans un espace abstrait, comme par exemple un espace 2D représentant la carrière professionnelle d'un individu (voir Figure I.8). L'entité mobile bouge dans cet espace dont les points sont des valeurs différentes d'un attribut complexe et variable dans le temps. Dans le cas de l'exemple de la figure I.8, c'est l'attribut " *carrière* ", défini par les coordonnées  $\langle \text{grade}, \text{institution} \rangle$ , qui varie. La variabilité peut être *discrète* (les valeurs sont existantes seulement à certains instants de temps), *stepwise* (les changements de valeurs peuvent être instantanés et chaque valeur est valable pour un intervalle de temps), et *continu* (les valeurs changent continuellement). À titre d'exemple, la trajectoire de la carrière professionnelle d'une personne est de type *stepwise*.

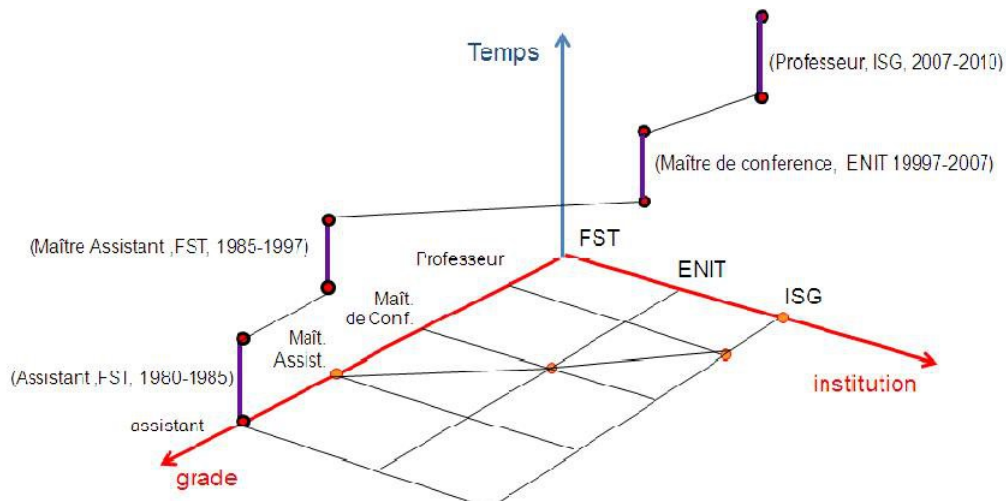
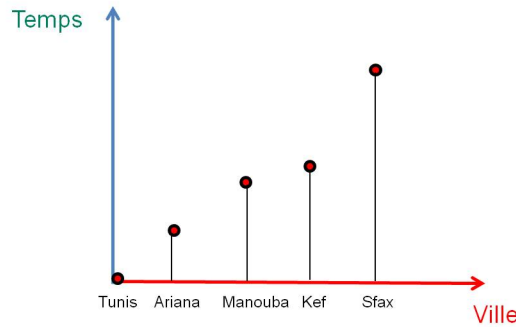


Figure I.8 – Exemple d'une trajectoire métaphorique représentant la carrière professionnelle d'un individu

#### Les trajectoires géographiques naïves

Dans ce cas, le déplacement s'effectue dans un espace géographique et ceci d'un endroit à un autre. Par contre, les positions géographiques ne sont pas définies ici en termes de coordonnées spatiales. À titre d'exemple, une trajectoire géographique naïve pourrait représenter le déplacement d'une ville à une autre dans un espace discret (voir Figure I.9). Les trajectoires géographiques naïves sont donc un cas particulier des trajectoires métaphoriques.



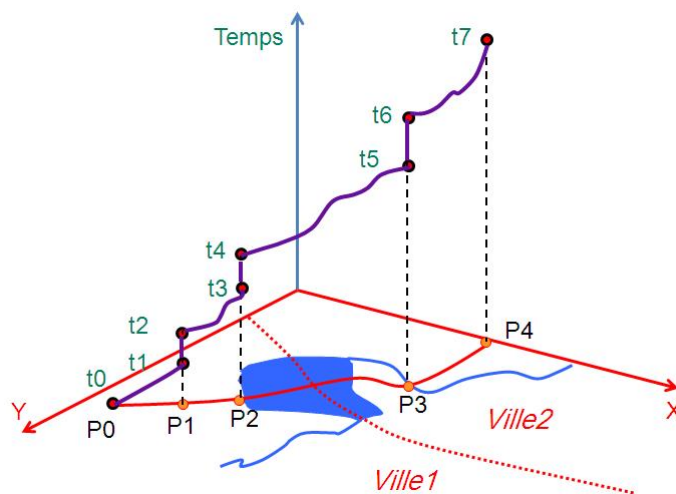


**Figure I.9** – Exemple d'une trajectoire géographique naïve représentant le déplacement entre villes

### Les trajectoires spatio-temporelles

Dans [76], une trajectoire est dite spatio-temporelle si la position d'un objet en mouvement subit des changements dans un espace géographique et qu'en plus les coordonnées spatiales sont utilisées pour exprimer cette position (voir Figure I.10). C'est ce type de trajectoires qui est habituellement géré par les bases de données d'objets mobiles.

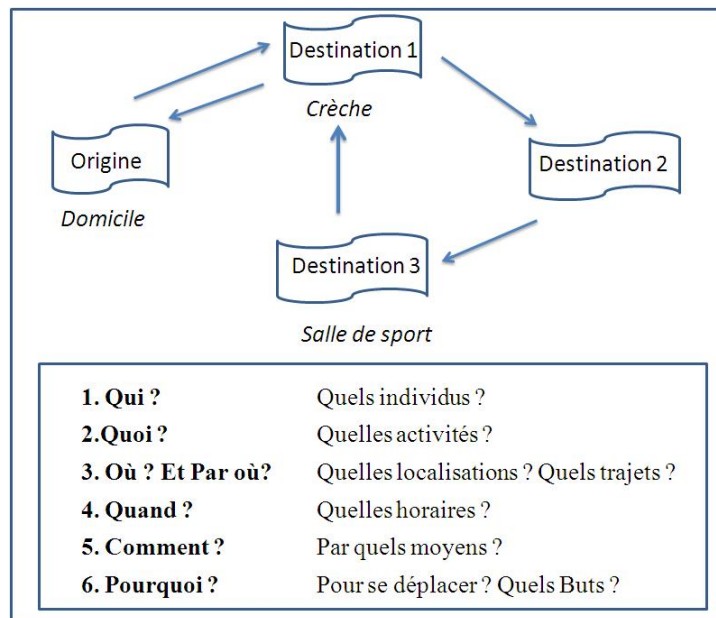
Il existe des termes synonymes des trajectoires spatio-temporelles comme le "*space-time path*" [29] ou les *lignes de vie géo-spatiales* (geospatial lifelines) introduites par [47]. En fait ces termes ne sont pas que des synonymes mais des cas spéciaux de trajectoires spatio-temporelles où l'intérêt porte sur la mobilité individuelle de personnes durant la totalité ou une partie de leur vie. Nous parlerons de ces trajectoires dans la section 3.5 car ils représentent un grand intérêt dans notre travail.



**Figure I.10** – Exemple d'une trajectoire spatio-temporelle

### 3.5 La trajectoire dans le cadre du Time Geography

Dans le cadre de l'étude du comportement de la mobilité quotidienne des individus, une dimension sémantique est rajoutée aux dimensions spatiales et temporelles du mouvement. En effet, un mouvement ou un déplacement n'est pas fini en soi ; un motif l'explique nécessairement même s'il correspond à une promenade ; "on ne se déplace pas pour se déplacer" [60]. Ainsi, l'étude de la mobilité quotidienne s'illustre à travers un chaînage d'activités et de déplacements (voir Figure I.11) prenant en considération les aspects spatiaux (Où ? par Où ?), temporeux (Quand ?) et sémantiques (Quoi ? Comment ? Pourquoi ?) [60].



**Figure I.11** – Mobilité quotidienne et chaîne d'activités et déplacements

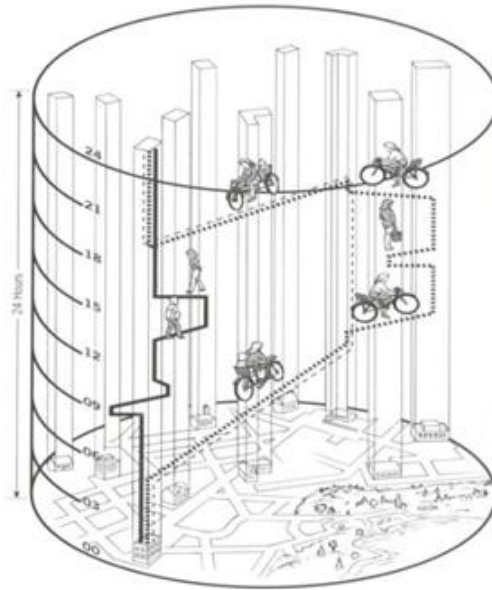
Avant les années soixante, aucun modèle reconnu n'associe ces différentes dimensions de la mobilité. C'est dans les années soixante-dix que Hägerstrand [78] propose un nouveau cadre de travail appelé "time geography". Dans ce cadre, l'espace et le temps fusionnent en une ressource indispensable à toute action, l'espace-temps [74]. La dimension temporelle est utilisée afin d'ordonner une séquence d'actions et de synchroniser les activités des individus. La dimension spatiale concerne les changements de localisations des objets. Le transport est considéré comme un moyen d'échange d'un budget temporel contre l'occupation d'un espace.

Dans cet espace bidimensionnel, chaque individu possède sa propre trajectoire (space-time path), séquence d'actions reliées par une chaîne de déplacements entre des lieux de stationnement (stations). Chaque trajectoire individuelle peut ainsi être représentée dans un espace à trois dimensions (une pour le temps, deux pour l'espace), par un cheminement spatio-temporel

## Chapitre I. Trajectoires spatio-temporelles d'objets mobiles

---

constitué de segments verticaux représentant la durée de chaque activité localisée dans l'espace (voir Figure I.12). La trajectoire spatio-temporelle dans le cadre du Time Geography a été aussi un "outil" pour l'étude de l'historique résidentielle d'individus durant les années vie [47]. Cette dernière étude a pour but de suivre les déplacements d'adresses d'habitation de personnes durant leur vie.



**Figure I.12** – L'idée de trajectoire spatio-temporelle

Nous donnons dans ce qui suit une description des composantes principales du cadre de Time Geography :

- Trajectoire spatio-temporelle (space time path) : Cette notion trace le mouvement d'un individu dans l'espace et dans le temps. Elle fournit une représentation spatio-temporelle explicite des différentes activités de l'individu, comme le temps de début ou de fin d'une activité, son emplacement, et de la séquence totale des activités. La trajectoire spatio-temporelle est composée d'une série connectée de segments inclinés et verticaux. Les segments inclinés indiquent le mouvement de l'individu dans l'espace et dans le temps. Les segments verticaux indiquent qu'il y a encore une consommation du temps mais qu'il y a une immobilité de l'individu dans l'espace. La pente d'un segment incliné représente la vitesse du déplacement.
- Prisme spatio-temporel (space-time prism) : Ce concept est une extension de la trajectoire spatio-temporelle [9]. Il définit un volume spatio-temporel délimitant les trajectoires possibles de l'individu. Il permet ainsi de mesurer la capacité des individus à atteindre

certains endroits dans l'espace et dans le temps. L'intérieur de ce prisme est appelé le *Potential Path Space*. La projection du prisme sur le plan géographique 2D est appelé *Potential Path Area*. Elle représente une région constituée de l'ensemble des emplacements géographiques que l'individu peut occuper durant un intervalle de temps  $[t_1, t_2]$  [29] (voir Figure I.13).

- Contraintes : Les trajectoires des individus sont contrôlées et influencées par des contraintes imposées par la société. De manière plus formelle, Tortsten Hagerstrand identifie trois types de contraintes :
  - Les contraintes de capacité (*capability constraints*) limitent notre " univers des possibles " pour des raisons physiologiques (nécessité de lieux de replis où manger, dormir), techniques (vitesse de déplacement) et topologiques (chaque possibilité de déplacement est circonscrite par une localisation donnée et à venir).
  - Les contraintes de conjonction (*coupling constraints*) recouvrent la nécessité de regrouper dans un même espace-temps (*bundle*) individus, outils et matériaux, dans un objectif d'activités de production et/ou de consommation et/ou d'interaction sociale. Ces activités sont fixes, tel que le travail si elles sont difficiles à re-ordonner ou à re-localiser. Elles sont dites flexibles, tel que le shopping, sinon. Les activités fixes dictent des contraintes de conjonction strictes, tandis que les contraintes flexibles permettent des conjonctions plus fluides dans le temps et dans l'espace.
  - Les contraintes de pouvoir (*authority constraints*) proviennent de l'organisation de l'espace en domaines (*domains*) hiérarchiquement emboîtés, qui trouve son origine dans la nécessité d'éviter que l'accumulation et la coexistence d'individus en un même lieu n'engendre des conflits de partage des ressources.
- Bundle : C'est l'illustration graphique de la convergence d'une ou plusieurs trajectoires spatio-temporelles ou la convergence d'une unique trajectoire avec une ou plusieurs ressources physiques, comme des équipements, des matériaux ou des bâtiments [36] (voir Figure I.13) . Cette notion prend tout son sens à des niveaux agrégés. En effet, le besoin de séquencer et de coordonner des tâches avec d'autres individus amène à la formation de bundles.
- Station : Dans une trajectoire, une station est l'endroit de formation des bundles. C'est un emplacement fixe avec une durée de temps limitée dans laquelle plusieurs trajectoires se joignent. Comme exemple, nous citons les écoles et les lieux de travail. Les stations sont habituellement illustrées sous forme de " tubes " (voir Figure I.14).

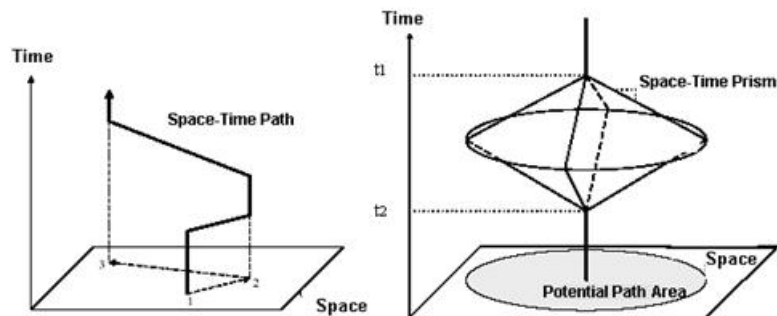


Figure I.13 – Trajectoire spatio-temporelle, prisme spatio-temporel et région de trajectoire potentielle

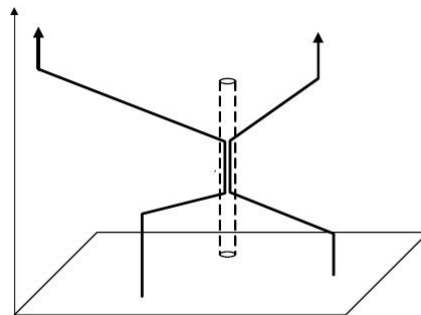


Figure I.14 – Les notions de *bundle* et de *stations* dans le time geography

## 4 Conclusion

Les trajectoires spatio-temporelles d'objets mobiles ont constitué, cette dernière décennie, une source d'information particulièrement importante. Cette nouvelle donnée, de part ses caractéristiques spatiales, temporelles et sémantiques, sert à quantifier le mouvement d'objets mobiles et à le caractériser. L'analyse efficace des trajectoires spatio-temporelles est devenue, donc, un élément primordial dans de plus en plus d'applications impliquant le mouvement de phénomènes complexes.

Dans ce chapitre, nous avons présenté les bases de données d'objets mobiles à travers les données manipulées dans ce type de base de données et à travers les applications qui les manipulent. Nous avons conclu, à travers cette étude, que la donnée gérée dans les bases de données d'objets mobiles sont les trajectoires spatio-temporelles.

Dans un deuxième temps, une étude de la notion de trajectoires spatio-temporelle d'objet mobile est présentée. Cette étude a permis de dégager les concepts de base des trajectoires et de mettre l'accent sur la caractéristique liée à la sémantique, qui est souvent ignorée dans

la littérature. Un intérêt particulier a été porté sur la présentation des trajectoires spatio-temporelles dans le cadre théorique du Time Geography, de part la sémantique qui se dégage dans cette représentation.

---

---

# Chapitre II

---

## Modélisation des trajectoires spatio-temporelles

### 1 Introduction

Avec le développement continu des technologies de capteurs de données en temps réel, et donc du potentiel lié à la modélisation d'objets mobiles, la communauté de recherche en bases de données spatio-temporelles s'est largement préoccupé ces dernières années du développement de modèles de représentation d'objets mobiles et de leur trajectoires.

La motivation principale de ces modèles est de définir, pour un objet qui se déplace de manière continue, une représentation sur laquelle on puisse construire un ensemble d'opérations intégrables à un langage d'interrogation. Les travaux existant dans la littérature varient quant à la perception des propriétés de l'espace (e.g., modèles de trajectoires libres ou modèles de trajectoires contraintes par le réseau) et du temps (e.g., le changement est traité selon un modèle discret ou continu). La problématique centrale de ces travaux a été en premier lieu la gestion de la continuité du mouvement. Plus récemment, la problématique de prise en considération d'une certaine forme de sémantique lors de la modélisation des trajectoires a vu le jour.

Nous donnons, dans ce chapitre, une présentation d'une panoplie des modèles, estimés les plus intéressants, de représentation de la trajectoire spatio-temporelle. Nous proposons pour cela, une classification de modèles de trajectoires spatio-temporelles qui différencie entre les modèles *avec apports sémantiques* et ceux *sans apports sémantiques*. Mais avant cela, nous identifions un ensemble de critères sur lesquels nous allons nous baser lors de la présentation de l'état de l'art des modèles de trajectoires spatio-temporelles et qui va nous servir pour positionner les modèles étudiés dans le paragraphe de la synthèse.

## 2 Critères d'évaluation des modèles de représentation des trajectoires

Le pouvoir d'expression d'un modèle pour la représentation d'une trajectoire spatio-temporelle doit s'apprécier du point de vue de différents critères comme la sémantique contenue dans cette représentation ou les possibilités de manipulation qu'il offre.

En se basant sur l'étude effectuée dans le chapitre précédent, et qui porte sur les trajectoires spatio-temporelles et leurs caractéristiques, nous fixons, dans ce qui suit, des critères que nous estimons importants et qui donnent quelques indicateurs pour une analyse des données trajectoires :

- **Approche de modélisation.** Les modèles de représentation des trajectoires spatio-temporelles d'objets mobile peuvent être décrits selon différentes approches. Les approches formelles, par exemple, utilisent dans leur représentation des formalismes mathématiques comme la spécification algébrique des types abstrait de données [71].
- **Référentiel spatial.** Une trajectoire spatio-temporelle peut être décrite dans un référentiel spatial basé sur la localisation ou à partir d'une représentation relative au sein d'un réseau. Dans le premier cas, les données spatiales de la trajectoire sont représentées par des coordonnées et/ou des primitives géométriques (e.g., série de points, polyligne) [58]. Dans le deuxième cas, une trajectoire est représentée à partir de la structure logique du réseau de transport et de ses contraintes, à savoir à l'aide d'arcs et de noeuds [86].
- **Type de mouvement/granularité temporelle.** La trajectoire peut représenter un mouvement de type continu et/ou discret. Le premier aspect donne une représentation plus proche de la réalité des objets mobiles et a fait l'objet de plusieurs travaux durant la dernière décennie [62]. Le second est intéressant, lorsqu'il n'existe pas de besoins de connaître avec précision les localisations des entités mobiles à tout instant de temps.
- **Sémantique.** Une trajectoire peut exprimer explicitement différents types de sémantique. Des informations dérivées des données spatiales d'une trajectoire, comme la direction ou la vitesse, sont considérés comme des informations sémantiques par certains [86]. Dans notre vision des choses, le critère sémantique représente plutôt la sémantique liée à la description explicite, au sein des trajectoires, de différentes informations telles que :
  - la mobilité et la non mobilité et les relations entre elles. Ceci peut se traduire par les notions de déplacements et d'activités introduits dans le cadre de Time Geography [78] ;
  - les raisons de la mobilité (Pourquoi se déplacer ? Quels buts ?) et les moyens utilisés (Comment se déplacer ?)



- les localisations ayant une signification particulière liée aux besoins des applications, comme par exemple les localisations des activités exercées aux endroits où il n’y a pas eu de mobilité. Nous nous intéressons, dans ce qui suit, à cette dernière forme de sémantique.
- **Possibilités de manipulation.** Un bon modèle de trajectoire doit permettre une manipulation aisée et efficace des trajectoires à différentes dimensions : spatiales, temporelles, spatio-temporelles et sémantiques. Idéalement, le modèle doit assurer une certaine orthogonalité dans cette manipulation ; dans le sens où l’utilisateur peut avoir la possibilité de manipuler chaque dimension séparément.
- **Généricité.** Cette propriété doit permettre de garantir l’indépendance du modèle de représentation d’une trajectoire vis-à-vis des domaines d’application (ou tout du moins une bonne flexibilité).

### 3 Modèles de trajectoires spatio-temporelles

Il existe dans la littérature différentes classifications des modèles de représentation des données spatio-temporelles. En effet, ces modèles peuvent être présentés selon l’un des critères proposés précédemment, comme par exemple le référentiel spatial adopté [86] ou l’approche de modélisation utilisée [15].

Par souci de clarté et de lisibilité de cette partie, nous présentons les modèles de trajectoires spatio-temporelles en utilisant certains critères parmi ceux que nous avons dégagés dans la section précédente. Vu que l’intérêt de notre travail porte essentiellement sur les critères liés à la sémantique du mouvement et l’approche de modélisation, nous classons les modèles de représentations des trajectoires spatio-temporelles selon qu’ils aient un apport sémantique ou pas. Pour chaque classe de modèle, nous différencions les modèles par approches formelles des modèles par approches non formelles.

#### 3.1 Modèles de trajectoires spatio-temporelles sans apport sémantique

Nous nous intéressons dans cette partie aux modèles ne présentant aucun enrichissement sémantique dans leur représentation de la trajectoires. Comme nous l’avons déjà mentionné dans la section 2, la sémantique désigne ici non pas les informations liées aux données brutes des trajectoires, mais plutôt celles liées aux besoins d’analyse des applications gérant des données mobiles.

Nous présentons dans ce qui suit une panoplie des modèles les plus importants dans la littérature en distinguant entre ceux qui adoptent des approches formelles de ceux qui adoptent des approches non formelles dans la représentation des trajectoires spatio-temporelles.

### 3.1.1 Modèles par approches formelles

Par souci de structuration de cette section, nous présentons ces modèles en les classant en deux classes selon le référentiel spatial utilisé.

**Modèles selon un référentiel spatial basé sur la localisation** L'un des modèles formels pionniers de la représentation des objets mobiles et de leur trajectoire est le modèle MOST (Moving Objects Spatio-Temporal data model), proposé par l'équipe de Wolfson [7, 61, 62]. Le modèle gère la position présente et future des trajectoires d'objets en mouvement continu. Pour ce faire, les auteurs proposent la notion d'*attributs dynamiques* dont la valeur change de façon continue dans le temps. Un attribut dynamique  $A$  est représenté par trois sous-attributs :

- $A.value$  : qui est la valeur de  $A$  au moment de la mise à jour,
- $A.updatetime$  : qui est le moment de la dernière mise à jour,
- $A.function$  : qui est une fonction de  $t$  qui vaut 0 à  $t = 0$ .

La valeur de l'attribut  $A$  au temps  $A.updatetime + t$  vaut  $A.value + A.function(t)$ . L'idée est qu'une mise à jour de la base de données n'est nécessaire que lorsque la position prédite de l'objet dévie de sa position actuelle ; autrement dit lorsque l'objet change de direction. À ce moment du changement, un *vecteur de mouvement*, relatif à l'objet et constitué de sa localisation, sa vitesse et sa direction durant la période du temps la plus récente, va être mis à jour par les nouvelles données au moment du changement. Les auteurs proposent aussi un langage de requête nommé FTL (Future Temporal Logic)[62]. Le langage proposé constitue une extension du langage de requête SQL par des prédicats de logique temporelle. L'approche proposée est restreinte aux points mobiles et ne permet pas de gérer les trajectoires de géométries plus complexes comme des régions mobiles. MOST a été implémenté dans un prototype nommé DOMINO (*Databases for MovINg Objects*)[63]. Le modèle représente la continuité du mouvement mais ne gère pas l'historique des trajectoires. Il est générique mais les capacités de manipulation qu'ils proposent se limitent à la dimension temporelle. Aucune sémantique du mouvement n'est exprimée dans cette représentation.

Dans [83, 84], une représentation relative des trajectoires permet d'appréhender les dynamiques de déplacement, d'objets géométriques, à partir d'une prise en compte des paramètres de vitesse et de position. La position relative est basée sur une notion de distance entre objets

mobiles, avec une distinction qualitative qui prend en compte la nature des relations topologiques entre ces objets. La vitesse relativisée est basée sur une notion de vitesse relative, où les objets plus rapides ou moins rapides que l'objet de référence sont différenciés [82]. L'association de ces mesures définit des états spatio-temporels caractérisés dans le langage naturel et composés afin de définir des relations transitives entre les objets. Le cadre expérimental de cette recherche est celui des trajectoires maritimes, à partir duquel est construit un modèle formel de représentation et de manipulation de trajectoires. Le formalisme proposé permet de caractériser le comportement d'un objet mobile selon le point de vue relatif d'un deuxième objet mobile. Le modèle permet de représenter des trajectoires continues dans le temps et intègre une forme de sémantique, liée à la notion de relativité d'un objet par rapport à un autre, qui ne correspond pas à l'objectif de notre travail.

Un travail fondamental sur la modélisation et la représentation de l'historique des trajectoires spatio-temporelles d'objets mobiles a été présenté dans [44]. Les auteurs proposent alors une nouvelle approche algébrique orientée *types abstraits de données* pour la modélisation et la requête d'objets mobiles. Le modèle proposé permet de représenter le développement temporel, continu et discret, d'entités spatiales de différentes géométries à l'aide de types abstraits de données comme le type *moving point* ou *moving region* [71].

Vu l'importance de ce travail, nous donnons dans ce qui suit une description détaillée de ces fondements. En effet, Le modèle proposé repose sur deux niveaux d'abstractions :

- le modèle "*abstrait*" qui repose sur un ensemble infini et fait abstraction des détails de l'implémentation,
- et le modèle "*discret*" qui propose une représentation finie qui implémente le modèle abstrait.

Le modèle discret proposé dans [71] est un système de type (voir Tableau II.1) décrit par une signature des types proposés et un ensemble d'opérations sur ces types (voir Tableau II.2). Le modèle se base sur un ensemble comprenant les types atomiques (*int*, *real*, *string*, *bool*,...), des types spatiaux pour représenter des données dans l'espace 2D (*point*, un point, *points*, un ensemble fini de points, *line*, une suite de courbes continues dans le plan, et *region*, ensemble fini de polygones disjoints) et enfin des types temporels (*instant*).

Les instances de ces types sont des valeurs statiques qui ne dépendent pas du temps. Des valeurs de types dynamiques pour des objets mobiles sont obtenues grâce au *constructeur de type moving*. Ainsi *moving(point)* est une instance d'un *point* mobile pouvant représenter une personne ou une voiture en mouvement.

Le principe de définition des opérations du modèle abstrait s'appuie sur un passage d'un ensemble générique d'opérations à un ensemble spécifique par application d'un processus de *lifting*

## II.3 Modèles de trajectoires spatio-temporelles

	→ BASE	<u>int</u> , <u>real</u> , <u>string</u> , <u>bool</u>
	→ SPATIAL	<u>point</u> , <u>points</u> , <u>line</u> , <u>region</u>
	→ TIME	<u>instant</u>
$BASE \cup TIME$	→ RANGE	<u>range</u>
$BASE \cup SPATIAL$	→ TEMPORAL	<u>intime</u> , <u>moving</u>

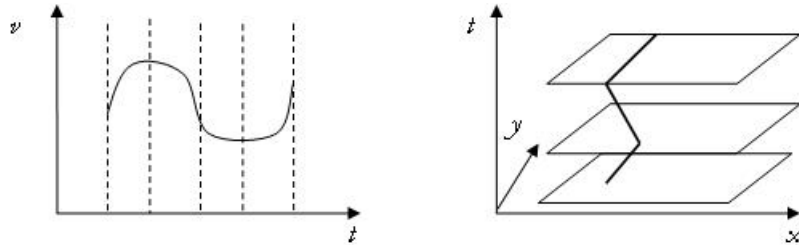
**Tableau II.1** – Signature décrivant le système de type abstrait de données pour la représentation des objets mobiles

Opération	Signature	
<b>trajectory</b>	$\underline{moving}(\underline{point})$	→ <u>line</u>
<b>length</b>	<u>line</u>	→ <u>real</u>
<b>distance</b>	$\underline{moving}(\underline{point}) \times \underline{moving}(\underline{point})$	→ <u>moving(real)</u>

**Tableau II.2** – Exemples de signatures de quelques opérations du modèle abstrait du modèle de représentation des objets mobiles par type abstrait de données

*temporel* vers un domaine spécifique. Le lifting temporel consiste à surcharger les opérations génériques définies sur les types non temporels (comme le type *point*) afin d'obtenir toutes les opérations possibles sur les variantes temporelles de ces types (comme le type *moving(point)*).

Dans [39], un système de type décrivant le modèle discret et une description de la représentation des types dynamiques *moving* sont données. Les auteurs expliquent que les valeurs des types sont obtenues grâce à une procédure de *mapping*. L'idée de base consiste à décomposer un déplacement d'objet en fragments appelés "tranches". Pour une tranche donnée, un mouvement peut être décrit, dans un référentiel basé sur les localisations, par une fonction du temps. Les représentations par tranches d'un réel mobile et d'un point mobile sont illustrées dans la Figure II.1.



**Figure II.1** – Représentation par tranches des valeurs d'un réel mobile et d'un point mobile

Afin d'étendre le modèle abstrait présenté dans [71], ces mêmes auteurs introduisent le concept de prédicats spatio-temporels [45]. Ce type de prédicats décrit le développement temporel de relations topologiques spatiales. Les auteurs décrivent un cadre dans lequel les pré-

dicats spatio-temporels peuvent être obtenus par agrégation temporelle de prédicats spatiaux élémentaires et par composition séquentielle. Sur la base de ces prédicats, un langage de requête spatio-temporelle, nommé STQL, est présenté [46]. Comme une dernière étape vers l'implémentation, Lema et al. [12] présentent une étude étendue d'algorithmes pour un large sous-ensemble des opérations proposées par [71]. Cette dernière contribution fournit un schéma pour l'implémentation d'extension de TAD aux objets mobiles. Ces extensions pourront être utilisées par des architectures de bases de données extensibles, par exemple une base de type relationnel-objet. Un prototype a été développé sous la forme d'un module algébrique pour le système expérimental de base de données SECONDO [72, 75] et comme un Datablade du SGBD Informix.

Un certain nombre d'autres travaux ont utilisé ou étendu l'approche proposée dans [71]. En effet, dans [69], une approche par type abstrait de données modélisant les trajectoires futures des objets en mouvement est proposée. Dans un contexte de datamining spatio-temporel, [51] s'appuie sur une partie du modèle de Gütting pour proposer un langage de requête de patrons spatio-temporels.

Une autre approche de modélisation par les TADs a été proposée sous le nom du projet Spatial TAU [53, 54]. Ce travail se base, aussi, sur le concept de types abstraits de données mobiles, précédemment décrits [12, 39, 71], pour concevoir un système de types mobiles sous la forme d'une cartouche de données ORACLE. D'un point de vue technique, deux cartouches de données sont implémentées. La première cartouche fournit des fonctionnalités purement temporelles en implémentant des types temporels selon le modèle TAU [30]. Les types temporels implémentés sont ceux définis par le standard objet ODMG [26] (Date, Time, Timestamp et Interval). De plus, le modèle définit trois nouveaux types temporels appelés : *Timepoint*, *Period* et *TemporalElement*. La seconde cartouche de données est le résultat de l'intégration des fonctionnalités temporelles avec les fonctionnalités spatiales offertes par la cartouche de données ORACLE Spatial. Ainsi, une panoplie de types mobiles est offerte à l'utilisateur (voir Figure II.2). Par exemple, le type générique *Moving\_Collection* permet de représenter des collections homogènes ou hétérogènes de types mobiles. Un ensemble de méthodes est proposé pour exprimer les processus et les phénomènes estimés intéressants. Cet ensemble de méthode sert à étendre le langage de requête PL/SQL à des sémantiques spatio-temporelles. Le système final proposé est une infrastructure de gestion de l'historique d'objets mobiles. Il a été évalué à l'aide d'un exemple d'une application de gestion du déplacement de camions. Plus tard, les mêmes auteurs ont étendus le système STAU à un prototype nommé HERMES [55–57] qui permet en plus de subvenir aux besoins d'applications réelles et temporellement dynamiques (tel que les Location-Based Services - LBS).

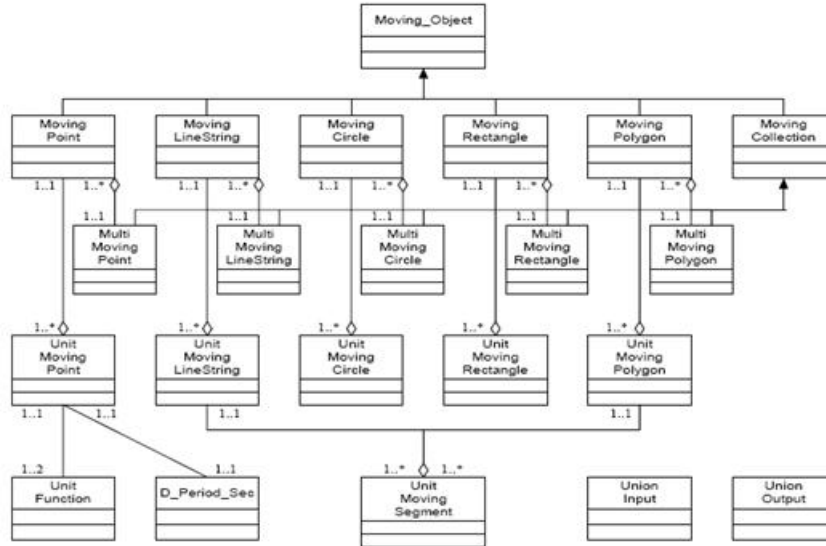


Figure II.2 – Diagramme de classe du projet STAU

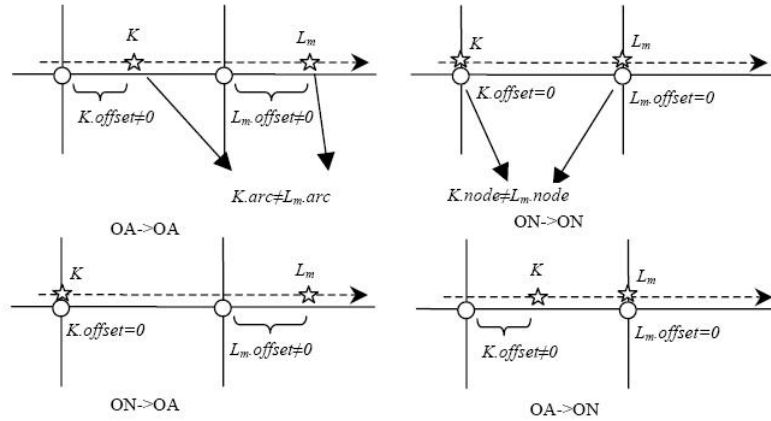
**Modèles selon un référentiel spatial basé sur le réseau** Li & al [86] proposent un modèle de représentation des trajectoires contraintes par une topologie de réseau. Le modèle proposé permet de caractériser et de sélectionner les points des trajectoires qui sont significatifs. Dans cette étude, une trajectoire contrainte par le réseau est modélisée comme un chemin logique constitué de séries ordonnées d'arcs et de noeuds. Sur la base de cette topologie, différents critères de sélection des points d'intérêts sont définis. À titre d'exemple, un des critères de sélection concerne les changements de relations entre un objet mobile et le réseau spécifique. Les auteurs définissent deux types de relations : un objet dans un arc (OA) et un objet dans un noeud (ON). Pour l'extraction des points clés, quatre types de changements sont distingués :

1. un déplacement d'un arc vers un noeud (OA->ON),
2. un déplacement d'un noeud vers un autre noeud (ON-> ON),
3. un déplacement d'un arc vers un autre arc (OA->OA)
4. et un déplacement d'un noeud vers un arc (ON->OA).

Les points clés sont générés lors de la détection d'un de ces types de changements entre deux points successifs de la trajectoire. La figure II.3 illustre des exemples des différents types de changements. Dans ces exemples,  $L$  est un point de localisation dans une trajectoire et  $K$  le premier point clé qui lui précède. Les deux autres critères de sélection sont définis sur la base des changements de direction et de vitesse des objets se déplaçant sur le réseau. L'approche

proposée a été implémentée dans un prototype étudiant le réseau de la ville de Hong Kong.

Bien qu'il introduit certaines relations, qualifiée de sémantique, facilitant l'analyse des caractéristiques des trajectoires sur un réseau urbain, le travail n'intègre aucune forme de sémantique liée au besoin des applications urbaines. De plus, aucun langage de requête ni de structures de données n'a été proposé pour l'interrogation et le stockage de des données mobiles.



**Figure II.3** – Types des changements des relations Objet-Réseau

Dans [27], un TAD pour des objets mobiles en réseaux a été présenté. Dans ce modèle, le système de type initialement proposé dans [71] a été enrichi par de nouveaux TAD. Il s'agit de combiner les approches de modélisation d'objets mobiles avec celles issues des réseaux de transport. Des types abstraits orientés réseaux comme *network*, *gPoint*, *mgPoint*, *gLine* et *mgLine* sont spécifiés. Le type *network* permet de représenter un réseau, non plus sous une forme d'arcs et de noeuds, mais sous une forme de route et de jonction entre ces routes. Les autres types proposés permettent de localiser un objet statique ou mobile sur un réseau. Sur la base du cadre présenté dans [71], un ensemble d'opérations spatiales, temporelles et spatio-temporelles sont proposées. Plusieurs opérations spécifiques au contexte de réseau, comme le calcul du chemin le plus court, sont également intégrées.

Avec une problématique cette fois intégrant la dimension des capteurs, [32] et [31] étendent le système de type proposé par [71] et [27] par des types représentant l'évolution de mesures, issues de capteurs mobiles, dans le temps et l'espace. Les auteurs proposent également deux nouvelles catégories de types : SVARIANT et GVARIANT. Ces catégories permettent de décrire la variabilité spatiale des mesures de capteurs des objets mobiles non contraints ou contraints par le réseau. Ces valeurs ne sont cependant pas directement intégrées dans la sémantique de la trajectoire, bien que leur variation en dépende fortement. Afin de profiter de la panoplie

d'opérations introduites dans [71], un mécanisme de *lifting spatial* est proposé. Comme pour le mécanisme de *lifting temporel*, le concept de *lifting spatial* consiste à surcharger les opérations disponibles au préalable pour les types non temporels pour les rendre applicables sur les nouveaux types.

### 3.1.2 Modèles par approches non formelles

Par souci de structuration de cette section, nous présentons ces modèles en les classant en deux classes selon le référentiel spatial utilisé.

**Modèles selon un référentiel spatial basé sur le réseau** Sur le plan des approches conceptuelles, [73] proposent une approche pour l'analyse, le pré-traitement, la modélisation et le stockage de données relatives aux trajectoires de véhicules se déplaçant sur un réseau. Nous estimons que ce modèle n'est pas générique car il restreint les objets mobiles aux objets de type véhicules. L'étape d'analyse complémente des travaux ultérieurs sur le même domaine [14]. En se basant sur l'étude du réseau routier de la région d'Athènes, cette étape a permis de dégager les spécificités spatio-temporelles et les concepts de base d'une base de données d'objets mobiles contraints par le réseau. Le concept central dégagé est la trajectoire. Ainsi, un ensemble de propriétés et de relations caractérisant la trajectoire de véhicules est proposé. Les propriétés les plus importantes concernent : la *vitesse* du mouvement, la *direction*, la *région* couvrant le déplacement des véhicules, la *distance* parcourue et le *temps de parcours*. Les relations dégagées dans ce modèle concernent des relations entre des trajectoires et des relations entre des trajectoires et leur environnement spatial. Une description conceptuelle orientée objet est proposée afin de dégager les caractéristiques du mouvement en organisant ses différentes composantes (voir Figure II.4).

Le modèle proposé permet de décrire la trajectoire des objets, en gardant trace de leurs positions sous la forme d'attributs spatio-temporels. Il gère donc un mouvement de type continu. Les auteurs notent que la représentation de ces positions peut être effectuée à l'aide des types abstraits de données mobiles proposés dans [71]. Un langage, nommé SML (Spatial Mining Language), de fouille de données mobiles est, au final, proposé. Néanmoins, l'expression de requêtes dans ce langage n'est pas toujours une tâche aisée. En effet, afin d'exprimer formellement et correctement la sémantique des requêtes, une parfaite connaissance du modèle proposé est nécessaire. Le modèle proposé a été implémenté comme le noyau d'un système de gestion de réseau routier, appelé " Path Finder ", ayant comme fonctionnalité principale la recherche de chemins parcourus par les véhicules dans le réseau routier de la région métropolitaine d'Athènes.



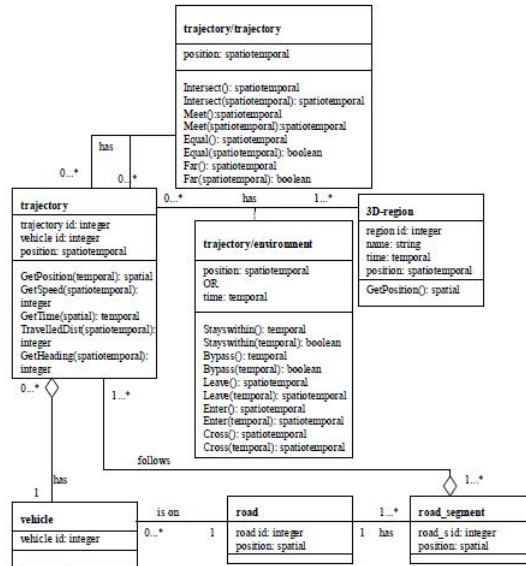


Figure II.4 – Conception orientée objet de véhicules mobiles sur un réseau routier et de leurs trajectoires

Plus récemment, [38] proposent une représentation de la trajectoire spatio-temporelle dans un contexte applicatif très particulier. En effet, le but de ce travail, n'étant pas la modélisation des trajectoires en elle-même, mais plutôt l'enrichissement sémantique d'une base de donnée routière à partir de trajectoires GPS de véhicules d'urgence. Pour cela, les auteurs décrivent une méthode, dont l'une des étapes est la modélisation des trajectoires continues mais tout en gardant la relation entre les trajectoires et le réseau physique. Les caractéristiques temporelles des trajectoires, comme la vitesse moyenne et le nombre et les temps d'arrêts sont utilisées dans la procédure d'enrichissement. Un TAD, qui adapte un type proposé par [27], est utilisé pour représenter la variation temporelle des vitesses des trajectoires.

**Modèles selon un référentiel spatial basé sur la localisation** Sur le plan des approches conceptuelles, Jin & al [66] proposent un CASE tool, nommé STXER, pour la modélisation conceptuelle des bases de données d'objets mobiles de type relationnel-objet. L'outil enrichit le modèle traditionnel Entité/Association avec des caractéristiques relatives à la mobilité d'objets de géométries variables (un incendie qui se propage) et d'objets de géométrie inchangée (une voiture qui se déplace). Il offre ainsi aux développeurs d'applications mobiles un pouvoir d'expression des différentes "sémantiques" spatio-temporelles de ces applications. Le modèle proposé permet de représenter le mouvement continu et discret ainsi que les changements des attributs ou des relations topologiques entre les objets mobiles et leur environnements. Aucun type de sémantique n'est pris en considération dans ce travail.

### 3.2 Modèles de trajectoires spatio-temporelles avec apport sémantique

Dans les travaux cités précédemment, la sémantique de la trajectoire d'objets mobiles est :

- soit inexistante,
- soit dérivable par des caractéristiques spatiales des trajectoires, pouvant être qualifiées de brutes, comme la position la vitesse ou la direction,
- soit dérivable à partir de la nature même du réseau (dans le cas où le déplacement est contraint par un réseau de transport),
- soit elle est caractérisée par un aspect "qualitatif" exprimant la relativité du déplacement d'un objet par rapport à un autre.

D'autre part, les auteurs de [13, 76] notent que les applications de gestion d'objets mobiles nécessitent une structuration plus efficace des données mouvement en plus d'une intégration d'une autre forme de sémantique. En effet, d'après les auteurs, d'un point de vue application, les connaissances sur les trajectoires inclut une multitude de données sémantiques qui peuvent compléter les enregistrements des positions spatiales des objets. À titre d'exemple, des applications de gestion de déplacement journalier d'employés pourrait bien s'intéresser aux modes de transport utilisés alors que des applications de surveillance de la migration de hordes d'oiseaux nécessiterait d'avoir des connaissances sémantiques sur les endroits, raisons et les durées des arrêts de ces animaux. De ce fait, de plus en plus de travaux récents se sont intéressés à l'enrichissement des trajectoires spatio-temporelles de ce type de sémantique. Nous exposons dans ce qui suit une panoplie de ces travaux.

#### 3.2.1 Modèles par approches formelles

Par souci de structuration de cette section, nous présentons ces modèles en les classant en deux classes selon le référentiel spatial utilisé.

**Modèles selon un référentiel spatial basé sur le réseau** Dans [36], Miller affirme la nécessité d'une approche formelle rigoureuse pour la représentation de la dynamique des objets spatiaux étudiés dans le cadre de la Time Geography. Les modèles étudiés sont pour la plupart des modèles conceptuels manquant d'un fondement mathématique solide. C'est dans cette perspective que, plus tard, Miller [29] propose un cadre formel, présenté comme une théorie de mesure analytique, dans lequel il définit mathématiquement les concepts (tel que le space time path) et certaines relations (tel que l'intersection) de base du cadre de Time Geography. Selon l'auteur, les définitions proposées sont assez générales pour traiter des espaces métriques de

dimension  $n$ , au lieu des espaces euclidiens bi-dimensionnels habituellement utilisés dans le cadre du Time Geography. Dans ce modèle, le concept de space-time path est présenté comme une composition de deux informations concernant les emplacements d'intérêts, à savoir les *points de contrôle*, et les chemins parcourus pour atteindre ces emplacements, à savoir des *segments de chemin* (*path segments*) reliant ces points.

Les points de contrôle sont, en effet, des points d'observations mesurés dans l'espace et dans le temps. Ils correspondent à différents types d'emplacements pouvant exister dans la réalité comme :

- des lieux d'activités connus (indiqué, par exemple, dans le journal d'activité de la personne),
- des emplacements où il y a eu un changement de direction ou de vitesse,
- des emplacements arbitraires (résultats habituellement des enregistrements des outils d'acquisition de données mobiles tel qu'un capteur GPS).

Le modèle proposé représente une séquence de ces points de contrôle comme un ensemble strictement ordonné dans le temps, permettant ainsi le parcours des localisations de la trajectoire en utilisant le temps comme indice. Les segments de chemin représentent, quant à eux, les localisations non observées dans l'espace et qui relient les points de contrôle temporellement adjacents. Ils sont modélisés comme des segments de lignes résultant de l'application d'une interpolation linéaire entre les points de contrôle. La vitesse moyenne du déplacement n'est pas supposée constante mais elle est plutôt déduite des points de contrôle.

Le modèle proposé par Miller offre une rigueur de formulation de la trajectoire dans le cadre du Time Geography. Néanmoins, nous constatons qu'aucune association n'est explicitement produite entre l'aspect sémantique et l'aspect spatio-temporel. Ainsi, le modèle ne permet pas de savoir par exemple à quels endroits se sont effectuées une certaine activité. D'autre part, le cadre n'offre aucune possibilité de manipulation de la trajectoire et ceci à aucune dimension. Le cadre reste théorique et aucune validation expérimentale, montrant l'intérêt de ce modèle sur des données spatio-temporelles, n'ont été proposées.

**Modèles selon un référentiel spatial basé sur la localisation** Ça n'est que très récemment, que Güting & al [37] a proposé un modèle générique de données mobiles qui prend en considération l'aspect sémantique. Le modèle énoncé enrichit le système de type, déjà proposé par l'équipe de Güting [71], par un ensemble de types modélisant le mouvement dans différents environnements (réseau routier, l'intérieur de bâtiments, espaces libres, etc.). De part sa généralité de modélisation, ce travail peut être considéré à la fois comme utilisant le référentiel spatial réseau et localisation. La sémantique représentée dans ce modèle se traduit par l'in-

## II.3 Modèles de trajectoires spatio-temporelles

---

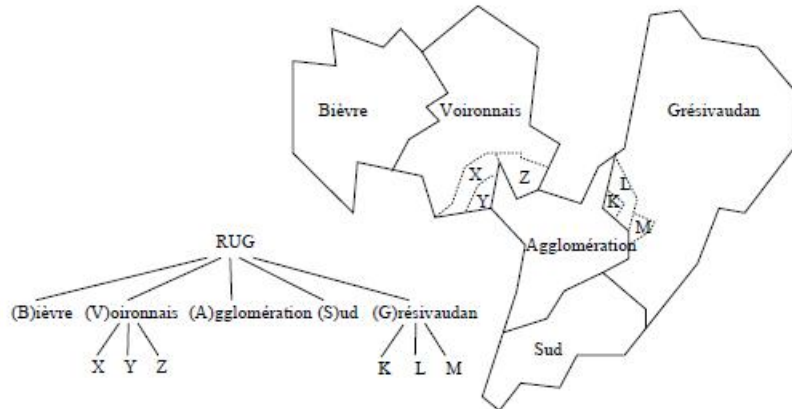
formation liée aux moyens de transports des entités mobiles qui est représentée au sein des types. Le système a été partiellement implanté dans le SGBD Secondo [75] et validé par un ensemble de requêtes émises en prenant en considération différents environnements. Bien que ce modèle soit robuste et malgré son aspect générique intéressant, nous estimons que l'information sémantique représentée est insuffisante aux besoins d'analyse et que les opérations de manipulations de l'aspect sémantique des types proposés sont très réduites. Ceci rend, à notre avis, les capacités d'utilisation et de gestion de l'aspect sémantique des types proposés pauvres et limite leur pouvoir de manipulation.

Dans le contexte d'une approche basée sur les activités, [74] proposent un cadre permettant la modélisation et l'interrogation des déplacements d'objets mobiles. Les objets étudiés, dits temporels, ont des états qui évoluent dans un espace spatio-temporel discret. Dans ce modèle, la trajectoire spatio-temporelle d'un objet est représentée sous forme de patrons de mobilité. Les auteurs définissent un patron de mobilité comme "*une forme récurrente apparaissant dans la succession des différentes valeurs prises par un (groupe d') individu(s) pour une dimension donnée (motivation, localisation, activité, etc.)*". Selon cette définition, on peut donc retrouver différents types de patrons : des patrons de déplacements, lorsqu'il s'agit de localisation, des patrons d'activités, etc. Plus spécifiquement, un patron de mobilité est une séquence paramétrée de symboles représentant les différents états d'un objet. Ces symboles sont pris dans un vocabulaire censé représenter l'ensemble des valeurs possibles par ces états. Un découpage multi-échelles des différents aspects de la mobilité (espace, activité, etc.) permet de fixer ce vocabulaire. Par exemple, étant donné une décomposition de l'espace en zones d'intérêts (voir Figure II.5), le patron X.Y.A.G représente tous les déplacements qui passent successivement dans les communes X puis Y dans le Voironnais, vont ensuite dans l'Agglomération, et finissent dans le Grésivaudan. L'aspect temporel peut aussi être représenté et il exprime le temps passé dans chaque zone. Selon les auteurs, donner à l'utilisateur le choix d'un niveau d'échelle particulier fournit un outil flexible pour agréger et comparer les déplacements.

Outre ce modèle de représentation, le travail définit également un langage d'interrogation de patrons. Une interrogation consiste à rechercher les objets dont la trajectoire correspond à un certain patron d'interrogation. Un tel patron consiste en une expression régulière comprenant des symboles de l'espace de référence, ainsi que des variables. Ainsi, la représentation d'une trajectoire prend la forme :

$$s_1\{T_1\}.s_2\{T_2\}.....s_n\{T_n\}$$

Où si,  $i=1..n$  sont les symboles de l'alphabet constitué lors du découpage de l'espace et  $T_i$  représentent le nombre d'unités de temps passées dans une zone  $s_i$ . Selon les auteurs, le



**Figure II.5** – Partition multi-échelles de l'espace pour l'élaboration d'un patron de mobilité

formalisme de patron proposé peut être facilement introduit dans le langage de requête SQL sous forme d'une extension géométrique. La syntaxe des expressions régulières est alors celle du langage Perl. Voici un exemple de requête retournant tous les objets qui se sont déplacés de la zone a à la zone f, qui sont restés au moins 2 unités de temps dans f puis qui sont partis vers la zone c :

```
SELECT *
FROM} Mob
WHERE matches(traj, 'a.f{2,.}c')
```

Dans cet exemple, la fonction *matches* vérifie si un suffixe de l'attribut spatio-temporel *traj* correspond au patron de mobilité 'a.f2,.c'. Le processus d'interrogation proposé permet aussi la manipulation de requêtes continues. Une requête continue est une requête dont le résultat est maintenu continûment pour un intervalle de temps donné. Des techniques et des algorithmes, basés sur les automates, ont été proposés pour l'évaluation de ce type de requête [11]. Une application web de suivi de voiture, appelée Sentinelle, a été réalisée dans le but d'évaluer le modèle proposé.

Bien qu'innovateur par son approche basée sur les patrons, ce modèle traite les comportements de déplacements et d'activités des individus d'une manière séparée. Autrement dit, le modèle propose deux types de trajectoires, l'une reliant l'espace et le temps mais sans précision sur la nature du déplacement et l'autre reliant les activités et le temps mais sans préciser l'espace. De plus, le chemin spatial qu'ont suivi les objets mobiles ne sont pas représentés dans ce modèle et les mécanismes de comparaison aux niveaux spatial, temporel et spatio-temporel sont réduits.

### 3.2.2 Modèles par approches non formelles

Par souci de structuration de cette section, nous présentons ces modèles en les classant en deux classes selon le référentiel spatial utilisé.

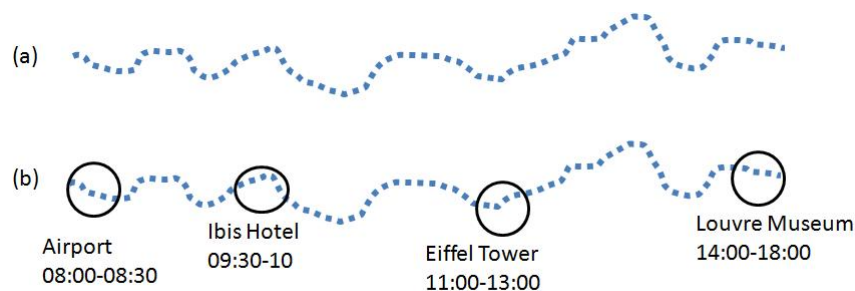
**Modèles selon un référentiel spatial basé sur la localisation** Dans [76], les auteurs proposent une approche qui aborde la notion de trajectoire comme un constructeur central sémantique. Cette méthode offre aux concepteurs de bases de données la possibilité de représenter la sémantique des données associées à la notion de mobilité, et en particulier celles des trajectoires identifiables. Les objets manipulés dans cette approche sont décrits, non plus comme des objets mobiles, comme précédemment, mais comme des objets ayant des trajectoires. Ce type d'objet peut posséder plusieurs trajectoires, définies temporellement par des intervalles de temps inclus dans la ligne de vie de l'objet. Afin de traduire cette idée, les auteurs proposent deux types de trajectoires : des *trajectoires simples*, définies sur des sous-ensembles temporels, et des *trajectoires complexes* qui sont des ensembles de trajectoires simples temporellement adjacentes ou disjointes. D'un point de vue spatial, le modèle utilise un référentiel basé sur la localisation. D'un point de vue temporel, les deux formes de représentation continue et discrète sont possibles. La continuité du déplacement est traduite sous forme d'une fonction du temps vers l'espace. L'ensemble des localisations de l'objet mobile est représenté afin d'assurer une représentation discrète de la trajectoire.

L'idée directrice de ce travail est d'allier une représentation des attributs génériques d'une trajectoire et de ses attributs sémantiquement liés aux applications. En effet, les auteurs dégagent des concepts généraux liés à la sémantique du mouvement. Ainsi, une trajectoire est composée, entre autres, d'arrêts (*stops*) et de déplacements (*moves*) ayant chacun un temps de début et un temps de fin (voir Figure II.6). L'interprétation et la définition de ces notions générales restent néanmoins à définir par les utilisateurs de l'application. D'un point de vue modélisation, les auteurs proposent deux solutions possible :

1. La première solution proposée consiste en un patron de conception (design pattern) qui explicite la représentation des trajectoires et de leurs composantes au sein d'un schéma de base de données.
2. La deuxième solution par contre, cache les données relatives aux trajectoires et consiste en un nouveau type de données *TrajectoryType* muni d'un ensemble de méthodes d'accès aux composantes des trajectoires.

Néanmoins, les relations liant les trajectoires à d'autres objets spatiaux, temporels ou spatio-temporels ne sont pas décrites par le type proposé. Les seules opérations décrites pour ce type

sont des opérations de calcul spatial, de recherche et d'acquisition de données. Sur la base de cette approche, l'équipe d'Alvares [59] propose une méthode de pré-traitement de données trajectoires afin de générer des données sémantiques tels que les localisations de la trajectoire, sémantiquement significatifs, où l'objet a passé une période minimale de temps. La méthode en question a été à la base d'un cadre de travail de pré-traitement de trajectoires spatio-temporelles [58] et d'un langage de requête relatif à la fouille de données trajectoires [79]. Plus récemment, la méthode d'Alvares a été implémenté comme un module d'extension spatio-temporel, appelé WAKA-STPM, d'une boîte à outils de fouille de données open-source appelée WEKA [81]. Le module proposé sert à accéder à une base de donnée spatio-temporelle et à ajouter des informations sémantiques aux trajectoires brutes qui y sont stockées.



**Figure II.6** – (a) Trajectoire brute (b) Trajectoire sémantique

Dans le même contexte, l'équipe de Spaccapietra [76] a approfondi sa recherche sur la modélisation de la sémantique des trajectoires spatio-temporelles. En effet, dans [87], une définition conceptuelle du concept de *trajectoire sémantique* est proposée et ceci dans le but de créer un framework pour l'annotation sémantique de trajectoire-spatio-temporelle. La trajectoire est modélisée comme une simple trajectoire brute enrichie par des annotations sur la sémantique du mouvement. Dans [13], Parent & al, proposent de nouvelles définitions sur différents concepts liés à la trajectoire et à son aspect sémantique e.g le comportement d'une trajectoire. Sur la base de ces concepts, Bogorny & al [80] proposent un modèle conceptuel de trajectoire et sous-trajectoires sémantiques, nommé CONSTnT, qui intègre, entre autres, ces nouveaux concepts (voir Figure II.7). Malgré l'intérêt de ce travail, nous notons qu'il n'offre aucune possibilité de manipulations de ces trajectoires ni à un niveau spatial, ni temporel, ni sémantique.

**Modèles selon un référentiel spatial basé sur le réseau** Dans le contexte de l'étude des interactions entre les activités humaines dans l'espace et dans le temps, selon Miller [29], la Time Geography offre un cadre théorique élégant et puissant pour l'étude des interactions des

## II.3 Modèles de trajectoires spatio-temporelles

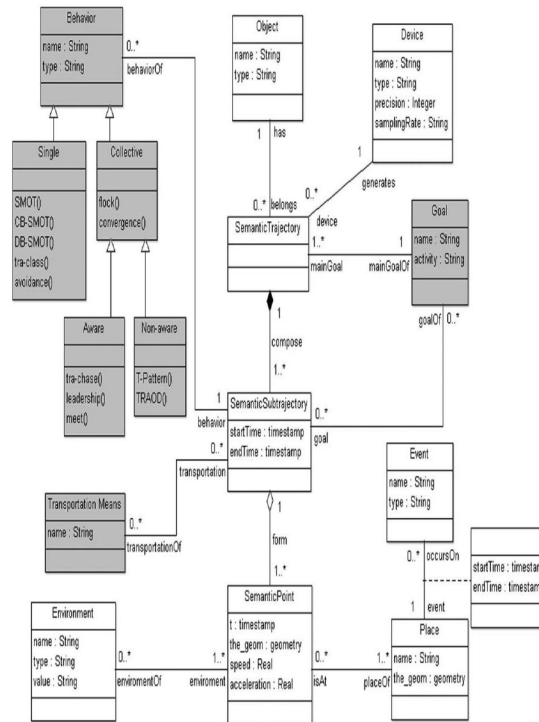


Figure II.7 – Modèle conceptuel pour la trajectoire sémantique

activités humaines dans l'espace et dans le temps. Ce cadre représente naturellement la sémantique de la mobilité à travers la représentation des activités (endroits d'arrêts pour l'exercice d'une activité) et déplacements d'objets mobiles dans l'espace et dans le temps à une certaine échelle d'observation. Nous nous intéressons donc dans ce qui suit aux principaux modèles de représentation de la trajectoire spatio-temporelle dans le cadre du Time Geography.

Partant de la problématique de la modélisation des concepts de Time Geography au sein d'une représentation basée sur le réseau, [28] présentent un cadre conceptuel pour la représentation et la visualisation de données issues de carnets de déplacements au sein d'un système d'information géographique spatio-temporel. Le modèle proposé organise les enregistrements relatifs aux déplacements conformément au concept de space time path. Ainsi, les caractéristiques spatiales et temporelles des déplacements d'une personne sont intégrées comme une classe d'attributs représentant sa trajectoire spatio-temporelle (voir Figure II.8).

Dans ce modèle, deux types d'objets sont représentés ; ceux qui n'ont aucune représentation géométrique (i.e Object Class) et ceux avec une représentation géométrique explicite (i.e *Feature Class*). Parmi ces derniers, les objets *Space-Time Path*, *ST Trip* et *Snapshot* sont des entités spatio-temporelles. Elles représentent respectivement : la trajectoire d'une séquence de déplacements organisés selon une séquence de temps, un déplacement (segment) de cette trajectoire



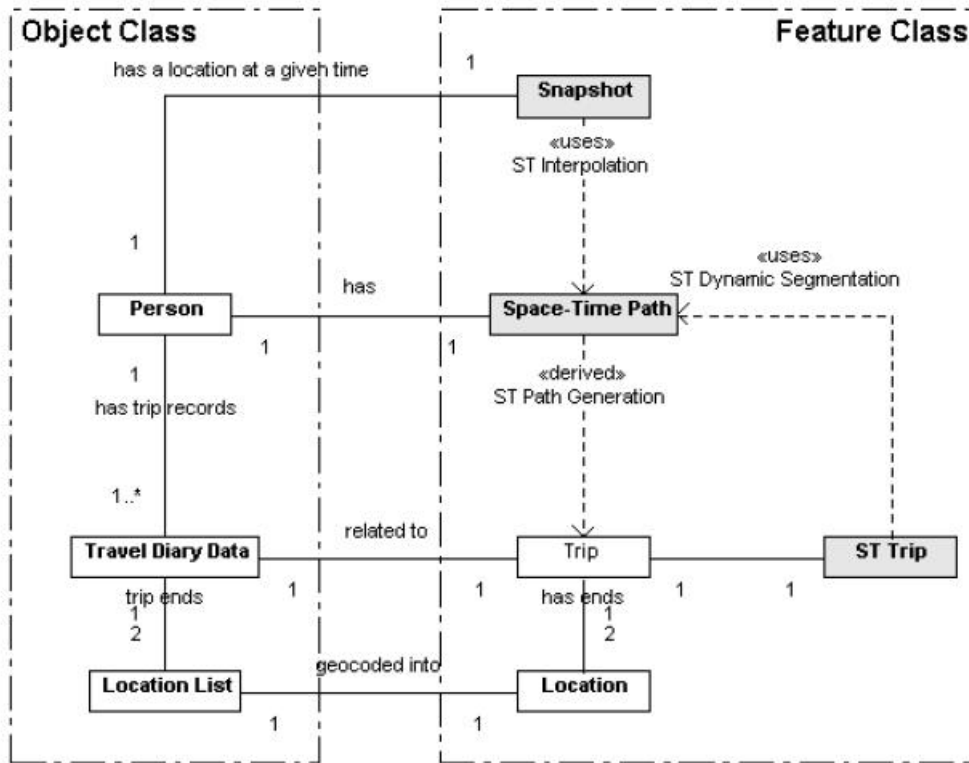


Figure II.8 – Modèle conceptuel pour la représentation de données issues de carnets de déplacements

et les localisations des personnes à des instants de temps différents. La dimension temporelle est représentée sous la forme d'attributs de classes. La classe centrale de cette représentation est la classe Space-Time Path. À partir de cette classe, les deux autres classes peuvent être générées par interpolation spatio-temporelle à des instants spécifiques et par segmentation dynamique. Une classe particulière *Travel Diary Data* encapsule les données relatives aux carnets de déplacements de personne et intègre donc une sémantique représentative du comportement de déplacements journaliers de personnes.

Même si les auteurs affirment avoir proposé un mode flexible et puissant pour l'exploration et l'analyse des trajectoires relatives au transport, ce travail reste néanmoins minimaliste sur plusieurs aspects. En effet, le modèle proposé représente le cadre spécifique des carnets de déplacements de personnes en déplacement. D'autre part, nous estimons que l'exploitation du concept de space-time path n'est pas optimale puisque la sémantique liée aux activités manque dans ce modèle. De plus, un effort reste à considérer concernant la représentation de la dimension temporelle. Nous notons aussi qu'aucunes relations caractérisant les trajectoires ni fonctionnalités d'extraction de trajectoires n'ont été définies, ce qui limite l'exploration de

## II.3 Modèles de trajectoires spatio-temporelles

ce modèle.

Dans le cadre de la résolution des problématiques liées aux demandes de transport basées sur les activités dans un environnement SIG, Wang and al [15] proposent un modèle conceptuel Entité/Association spatio-temporel (voir Figure II.9). Dans ce travail, les dynamiques des programmes d'activités sont représentées d'une façon discrète, plus précisément en adoptant un type de comportement *stepwise* reflétant le fait qu'un individu est soit dans un état *statique*, soit en *mouvement*. La temporalité est implémentée par :

- une séquence ordonnée de paire d'attributs spatio-temporels composés (*Stay\_AT* et *Travel\_Between*) correspondant respectivement à l'état statique et dynamique d'un individu,
- des relation *before* et *after* qui permettent la navigation à travers la topologie des déplacements (*Travel\_Between*) et des activités (*Stay\_AT*).

Bien que ce modèle représente la globalité des concepts et sémantique du domaine d'application, il reste limité quant aux opérateurs topologiques spatiaux, temporeux offerts et quant à la capacité d'expression de requêtes spatio-temporelles complexes.

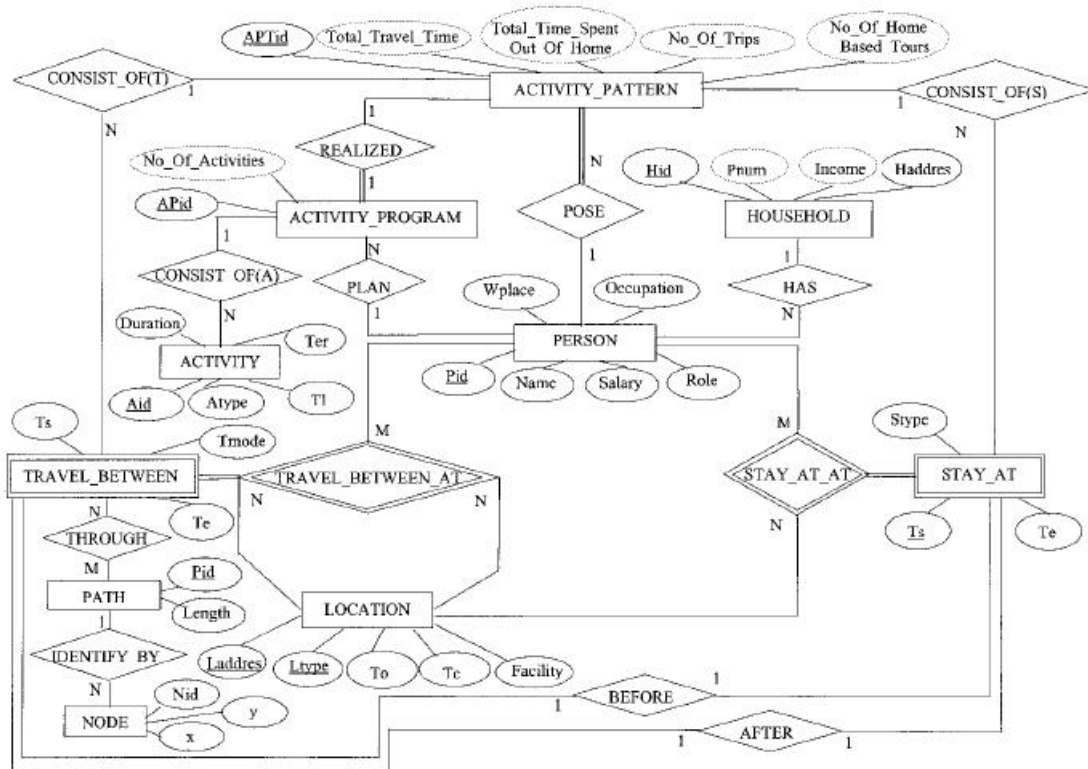


Figure II.9 – Diagramme entité-association d'un modèle de données basé sur les activités

Dans le cadre de la modélisation des comportements individuels de mobilité basés sur l'ac-

## Chapitre II. Modélisation des trajectoires spatio-temporelles

tivité, [6] proposent un modèle de données spatio-temporelles orienté objet (voir Figure II.10). Ce modèle conceptuel permet de représenter les événements significatifs pour un échantillon de personnes qui ont répondu à une enquête origine-destination (O-D). Implanté dans un SIG temporel, le modèle conceptuel relie les individus, leurs déplacements, les itinéraires suivis, leurs activités ainsi que les ménages concernés avec les infrastructures de transport (réseaux à structure topologique) ainsi que les lieux de résidence et d'activité situés dans la région métropolitaine de Québec, au Canada. Un prototype de base de données a été obtenu en implantant ce modèle de données dans le logiciel Smallworld GIS.

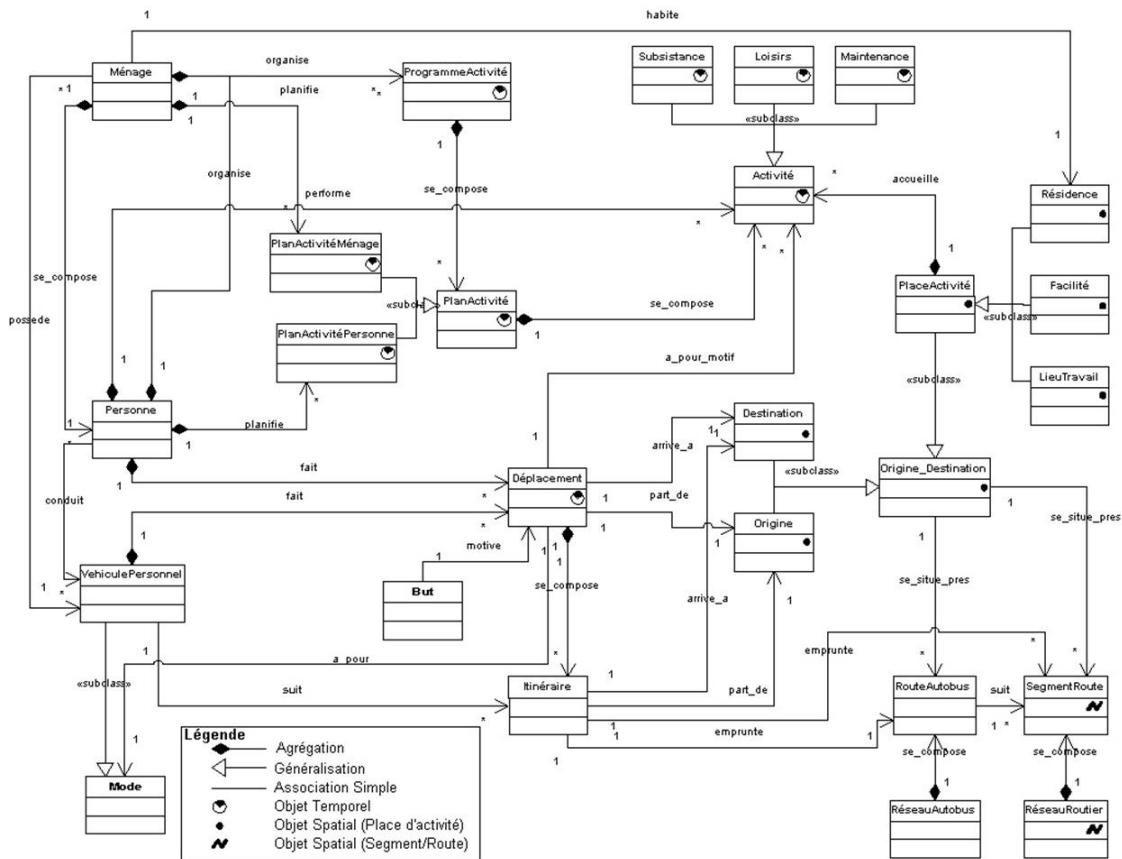


Figure II.10 – Modèle de données spatio-temporelles orienté-objet basé sur les activités et déplacements

Dans la seconde étape du projet [5], une dimension temporelle a été ajoutée au prototype. En considérant le modèle de temps linéaire tel qu'il est enchâssé dans les données de l'enquête O-D, les auteurs ont défini l'événement temporel atomique du comportement individuel de mobilité basé sur l'activité. Ainsi, chaque événement (activité ou déplacement) est un intervalle temporel délimité par un temps de début et un temps de fin. Cet événement atomique est utilisé pour générer les enchaînements d'événements de chaque type : activité ou déplacement. Ces

enchaînements constituent des ensembles totalement ordonnés. L'union de l'enchaînement des activités et celui des déplacements produit la trajectoire temporelle sur laquelle les activités et les déplacements individuels s'intercalent.

La définition de l'événement atomique a été insérée comme attributs temporels dans les objets Déplacement et Activité. Afin de tirer profit de cette nouvelle extension, les auteurs ont développé un moteur de requête temporel implémentant différents prédicats temporeux. Ce moteur permet de traiter des requêtes de sélection standard, basées sur les prédicats d'Allen [23]. Il traite aussi des requêtes sur les chaînes d'événements. Ce type de requête permet de rechercher un événement spécifique sur la base de sa position dans l'ordre de chaînage. Ceci est effectué à l'aide d'opérateurs de position, conçus pour transformer des requêtes topologiques en des fonctions de recherche dans les ensembles, basées sur les indices. Ce genre d'opérations de catégorie sémantiques et temporelles, bien que riches, sont les seules opérations de manipulation de la trajectoire spatio-temporelle dans ce modèle. Il n'existe en effet, aucune possibilité de manipulation des dimensions spatiales et spatio-temporelles de la trajectoire.

## 4 Synthèse

Dans la section précédente, un certain nombre de modèles de trajectoires spatio-temporelles ont été étudiés et classés selon les critères que nous avons préalablement fixés. Dans cette synthèse, nous positionnons ces mêmes modèles par rapport à ces critères. Le tableau II.3 illustre ce positionnement et présente, donc, une comparaison entre ces différents modèles.

Notre état de l'art sur les modèles de trajectoire spatio-temporelles, montre que la problématique de la représentation de la continuité du mouvement a été largement abordée par la majorité des travaux existants, comme [62, 71].

En ce qui concerne les approches de modélisation, nous constatons que les modèles qui adoptent des approches conceptuelles sont souvent liés à un contexte d'application particulier. À titre d'exemples, dans [38] le contexte de modélisation est restreint aux trajectoires de véhicules d'urgences alors que dans [73] seuls les véhicules sont pris en considération comme objets mobiles.

D'autre part, toujours en terme d'approche de mobilisation, notre état de l'art montre que les approches formelles offrent des représentations qui se distinguent par leur rigueur dans la représentation des trajectoires spatio-temporelles. En particulier, l'approche par type abstrait de données a eu beaucoup de succès lors de la modélisation des bases de données spatio-temporelles. Certains travaux proposent des types abstraits de données pour les objets mobiles [37, 54, 71],

d'autres viennent compléter ces représentations ou les utiliser dans leur représentations [51, 69]. Cet engouement est dû à la rigueur de formalisation de cette approche, sa facilité d'intégration dans une base de données et pour l'indépendance de modélisation, qu'elle assure, d'un type particulier de base de données (e.g. relationnelle, objet, etc.).

Certes, cette méthode ne garantit pas la souplesse et la facilité d'extension mais ses avantages sont intéressants. Ceci est vrai, d'autant plus que nous pensons que, vu l'importance de la donnée trajectoire spatio-temporelle dans les bases de données spatio-temporelles, il est de plus en plus nécessaire de percevoir cette donnée comme une entité à part entière, identifiable et aisément manipulable; et c'est exactement ce qu'offre la modélisation par type abstrait de données.

Néanmoins, en terme de représentation de la sémantique des trajectoires telles que nous l'avons modélisée, nous avons remarqué que très peu de modèles de trajectoires spatio-temporelles par types abstraits de données parmi ceux que nous avons recensés permet d'exprimer une partie de cette sémantique [37]. Dans le modèle de [37], seule une information sur les moyens de déplacements d'un objet mobile est intégrée comme donnée sémantique et les opérations de manipulation de cette sémantique sont quasi inexistantes.

D'autre part, le cadre de Time Geography offre un cadre conceptuel puissant pour représenter naturellement les dimensions spatiales, temporelles et sémantiques des trajectoires. En effet, les trajectoires, dans ce cadre, représentent parfaitement des phénomènes de déplacements et d'activités. Cette représentation offre un lien sémantique entre le mouvement (se déplacer) et sa sémantique (pourquoi se déplacer?, etc.) qui permet de modéliser adéquatement plusieurs phénomènes comme les migrations de hordes d'animaux, la propagation de virus, les déplacements d'individus et de véhicules, les mouvements de troupes dans un champ de bataille et éventuellement la ligne de vie d'une entité.

Néanmoins, l'état de l'art a montré une insuffisance de modèles de trajectoires qui s'inspirent de ce cadre et surtout un manque de formalisation de ses concepts de base, en particulier la notion de "*space time path*". D'autre part, le peu de modèles formels de trajectoires dans ce cadre qui existent, comme [29], offrent des possibilités limitées et restreintes de manipulations de ces trajectoires.

Afin de compléter les représentations actuelles, notre but est de remédier aux deux problématiques précédemment citées. Nous avons donc comme objectifs de proposer un modèle formel de trajectoire spatio-temporelle qui prend en considération l'aspect sémantique de la mobilité. Le modèle proposé définira formellement un type abstrait de données pour la trajectoire spatio-temporelle, nommé TST, qui devra :

- être rigoureusement formalisé;

- être facilement intégré dans un environnement spatio-temporel tel qu'un SGBD spatial ;
- être indépendant d'un type particulier de base de données (e.g les bases de données relationnelles) ;
- être doté d'un ensemble riche d'opérations de manipulations à différentes dimensions, spatiales, temporelles, spatio-temporelles et sémantiques ;
- s'appuyer sur le cadre de time geography dans sa représentation de la trajectoire spatio-temporelle ; cette représentation, en effet, associe l'aspect spatio-temporel et la mobilité à son aspect sémantique.

Notre modèle devra par la suite être mis en oeuvre au sein d'un SGBD spatial et devra être expérimenté sur des données benchmark dans le contexte d'un cas d'étude qui sera préalablement choisi.

Modèle	Approche de modélisation	Référentiel spatial	Type du mouvement	Possibilités de manipulation	Sémantique	Généricité
<b>Modèles sans apport sémantique</b>						
MOST [61]	formelle	localisation	continu	une logique de prédiction temporelle (FTL)	non	oui
Représentation relative des trajectoires [83]	formelle	localisation	continu	non	non	oui
TAD pour les objets mobiles [71], [57] [69] [51]	formelle par TAD	localisation	continu	spatiales, temporelles, spatio-temporelles	non	oui
Représentation des trajectoires par points clés [86]	formelle	réseau	discret	non	non	oui
TAD d'objets mobiles sur le réseau [27]	formelle par TAD	réseau	discret	spatiales,temporelles, spato-temporelles	non	oui
Modèle conceptuel de véhicules mobiles [31]	formel par TAD	réseau	discret	spatiales,temporelles et spatio-temporelles dérivées de [71]	non	non
Modèle pour l'enrichissement des trajectoires de véhicules d'urgences [38]	conceptuelle	réseau	continu	spatiales et temporelles	non	non
Modèle conceptuel pour le déplacement urbain de véhicules [73]	conceptuelle	réseau	continu	spatiales, temporelles, spatio-temporelles dérivées de [71]	non	non
Case tool STXER [66]	conceptuel	localisation	continu et discret	non	non	oui
<b>Modèles avec apport sémantique</b>						

Modèle	Approche de modélisation	Référentiel spatial	Type du mouvement	Possibilités de manipulation	Sémantique	Généricité
Modèle formel pour la trajectoire spatio-temporelle dans le Time Geography [29]	formel	localisation	continu	non	oui mais n'exprime pas les liens entre les concepts sémantiques dans sa représentation	oui
TAD d'objets mobiles multi-environnements [37]	formel par TAD	localisation et réseau	continu	spatiales, temporelles, spatio-temporelles mais très peu d'opérations sémantiques	oui mais pauvre	oui
Patrons de mobilité [74]	formel	localisation	discret	non	oui mais les liens entre les concepts de déplacements et activités sont indépendants	oui
WAKA-STPM [76], [59], [13], [80]	conceptuel	localisation	continu	non	oui	oui
Modèle conceptuel O.O de représentation des carnets de déplacements d'individus [28]	conceptuel	localisation	continu	non	oui	non
Modèle conceptuel E/A des programmes d'activités d'individus [15]	conceptuel	réseau	discret	non	oui	non



Modèle	Approche de modélisation	Référentiel spatial	Type du mouvement	Possibilités de manipulation	Sémantique	Généricité
Modèle conceptuel O.O des programmes d'activités d'individus[6]	conceptuel	réseau	discret	temporelles et sémantiques	oui	oui

**Tableau II.3** – Tableau comparatif des modèles de représentation de la trajectoire spatio-temporelle

## 5 Conclusion

Dans ce chapitre, nous avons proposé un état de l'art sur les modèles de trajectoires spatio-temporelles en s'appuyant sur des critères préalablement définis. Une étude comparative des différents modèles a été, par la suite, présentée. Il en ressort de cette étude, d'une part l'intérêt des modèles formels de représentation des trajectoires et d'autre part le manque de modèles formels intégrant la dimension sémantique dans leurs représentations. Les deux prochains chapitres décrivent le modèle formel par type abstrait de données et sémantiquement riche que nous proposons et que nous nommons TST.

## Deuxième partie

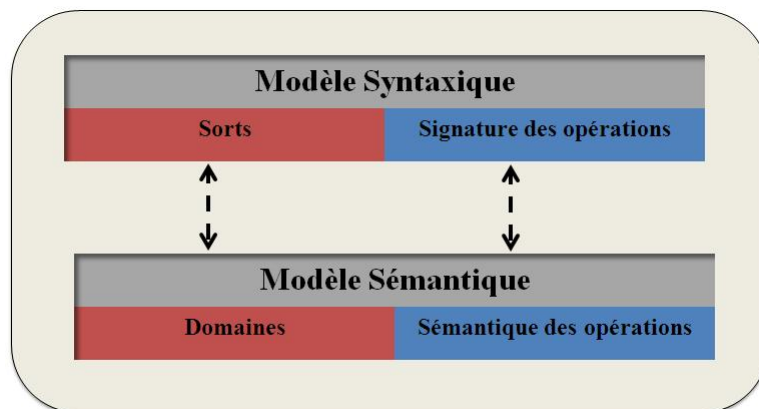
**TST : un type abstrait de données  
pour les trajectoires spatio-temporelles  
sémantiquement riches**



Nous proposons dans cette partie un modèle de représentation des trajectoires d'objets mobiles ainsi qu'une validation expérimentale de ce modèle. Ce modèle est basé sur une approche algébrique par type abstrait de données (TAD) et offre une couverture de représentation des différentes caractéristiques de la trajectoire, particulièrement la dimension sémantique.

Un nouveau type de donnée, nommé TST, représentant les trajectoires spatio-temporelles est proposé ainsi qu'un ensemble d'opérations de manipulation du TST. La finalité est que le type TST soit mis à la disposition des concepteurs des bases de données spatio-temporelles comme tout autre type de base et qu'il puisse être manipulé efficacement à différentes dimensions, spatiales, temporelles, spatio-temporelles et sémantiques. Le modèle est proposé tout en gardant à l'esprit les objectifs définis au préalable, à savoir une rigueur de formalisation, une indépendance de la représentation d'un modèle de base de données particulier et une richesse opérationnelle du type.

La spécification algébrique d'un TAD consiste à définir une syntaxe, sous la forme d'une signature et d'associer à cette signature une sémantique formellement définie [33] (voir Figure II.11).



**Figure II.11** – Les composantes de modélisation du type TST

La signature est un couple  $(Sorts, \Omega) \in Sorts$  :

- *Sorts* est l'ensemble des types nécessaires pour définir le type en question,
- $\Omega$  est la liste des opérations qui lui sont associées.

La sémantique sert à décrire les données utilisées par le type et le comportement des opérations de la signature par des mécanismes formels, comme des axiomes, des lois ou des fonctions d'interprétations.

Pour la modélisation du nouveau type TST, nous utilisons la méthode de spécification dénotationnelle [77]. Cette méthode décrit la sémantique d'une algèbre, non plus à travers une axiomatique, mais à travers un modèle mathématique concret. Alors qu'une spécification

---

axiomatique décrit le sens du type en fonction de son impact sur certaines propriétés, la méthode dénotationnelle définit la sémantique du type par un objet mathématique précisément défini (c'est la *dénotation* du type). Dans notre modélisation, cet objet est une fonction mathématique des entrées vers les sorties, plus précisément de domaines vers d'autres.

La méthode dénotationnelle possède plusieurs avantages. En effet, elle apporte à la sémantique du TAD une rigueur, non ambiguïté et précision de formulation et une indépendance de la description d'un modèle précis de SGBD. D'autre part, cette méthode peut être utilisée comme définition d'un type dans un but de standardisation.

Nous modélisons, par la suite, le type TST en deux étapes. En premier lieu, nous donnons, dans le chapitre III, une spécification formelle de ce type. Cette spécification représente la *signature* ou le *modèle syntaxique* du type proposé. Dans le chapitre IV, l'algèbre liée à cette signature est spécifiée. L'algèbre proposée représente le *modèle sémantique* de notre type.

La validation de ce modèle se traduira, dans le chapitre V, par une mise en oeuvre du type TST et de ses fonctionnalités au sein d'un SGBD spatial ainsi qu'une expérimentation.

---

---

# Chapitre III

---

## Modèle syntaxique du TST

### 1 Introduction

L'expression de la syntaxe d'un type abstrait de données se traduit par l'expression de la syntaxe des données à modéliser et des mécanismes de manipulations de ces données. Afin de proposer une syntaxe adéquate aux besoins de modélisation d'un type abstrait des données trajectoires spatio-temporelles, nous commençons dans ce chapitre, par une exposition du contexte spatial et temporel du modèle proposé. Ceci fera l'objet de section 2.

D'autre part, et afin de pouvoir proposer un aspect opérationnel riche qui recouvre les principaux aspects de la trajectoire, nous dégageons, dans la section 3, quelques besoins d'analyse de la trajectoire spatio-temporelle d'objets mobiles ; besoins que nous comptons couvrir au mieux dans notre proposition.

Enfin, nous proposons le modèle syntaxique du type TST dans la section 4. Le modèle est composé de deux parties :

1. *Les sorts* : Il s'agit de fixer les types et données utilisées pour la description du type TST.
2. *La signature des opérations* : Il s'agit de proposer les opérations de manipulation du type TST en définissant leur signature.

### 2 Définition de notre contexte de modélisation

#### 2.1 Contexte spatial et temporel

Avant de présenter notre modèle, nous donnons tout d'abord quelques précisions sur le contexte spatial et temporel de notre travail. Autrement dit, nous présentons ici les propriétés de l'espace et du temps dans lesquels nous supposons que le mouvement a eu lieu.

### 2.1.1 Propriétés spatiales

L'espace ici est, comme déjà mentionné dans le chapitre I, un espace géographique dont les propriétés sont les suivantes :

- **Référentiel spatial** : Le système de référence que nous allons adopter pour décrire la trajectoire spatio-temporelle est *un système basé sur la localisation*. Les localisations seront donc représentées par des primitives géométriques, tels qu'un *point* spatial ou une *polyligne*.
- **Espace continu/discret** : L'espace, ici, est de nature continue.
- **Dimension** : La dimension utilisée dans notre représentation est une représentation bi-dimensionnelle ; c'est à dire qu'une localisation est un point spatial défini par deux coordonnées.
- **Échelle spatiale** : Notre modèle ne se restreint pas à une échelle spatiale particulière. Le choix de l'échelle spatiale appropriée dépend des besoins de l'application et reste donc la tâche de l'utilisateur.
- **Relations spatiales** : Nous supposons ici travailler dans un espace géographique muni de relations topologiques entre entités spatiales.

### 2.1.2 Propriétés temporelles

Les propriétés du temps dans notre modélisation sont les suivantes :

- **Nature du temps** : L'approche du temps utilisée dans notre travail est une approche *newtonienne*.
- **Entité temporelles** : Nous allons utiliser pour la description de la trajectoire spatio-temporelle les primitives *point temporel* (*instant de temps*) et *intervalle de temps*. Les intervalles sont supposés fermés.
- **Topologie du temps** : Nous adoptons une représentation linéaire avec une relation d'ordre total.
- **Relations temporelles** : Nous supposons travailler dans un modèle temporel muni d'opérations de topologies temporelles entre points et intervalles [23].
- **Granularité temporelle** : Comme pour la granularité spatiale, la granularité temporelle n'est pas fixée dans notre représentation de la trajectoire mais elle est plutôt laissée comme un choix de l'utilisateur.
- **Temps valide ou temps de transaction** : Le temps utilisé est le temps valide ; c'est à dire le temps d'occurrence du fait dans la réalité.
- **Structure/densité temporelle** : La structure du temps utilisée est une structure dis-



crète. Le temps est donc mesuré à des instants particuliers et les points temporels sont isomorphes à des entiers naturels.

### 2.2 Définitions

Précédemment, au niveau du chapitre I, nous avons présenté la notion de trajectoire spatio-temporelle dans le cadre du Time Geography. Dans ce cadre, la conception de la trajectoire est un chaînage alternatif de déplacements et d'activités se succédant dans le temps. Ainsi, chaque déplacement, effectué à l'aide d'un mode de déplacement, est associé à un arrêt qui peut représenter sémantiquement l'objectif de ce déplacement (pourquoi se déplacer ?) qui est au final le déroulement de l'activité correspondante. Cette structuration de la trajectoire, qui permet une association entre le déplacement et sa sémantique, nous semble offrir un cadre riche et approprié pour la représentation spatio-temporelle et sémantique de la trajectoire d'objets mobiles.

Dans notre travail, nous tirons profit de ces concepts pour la représentation de notre modèle. Pour cela, nous donnons les définitions suivantes.

#### 2.2.1 Activité

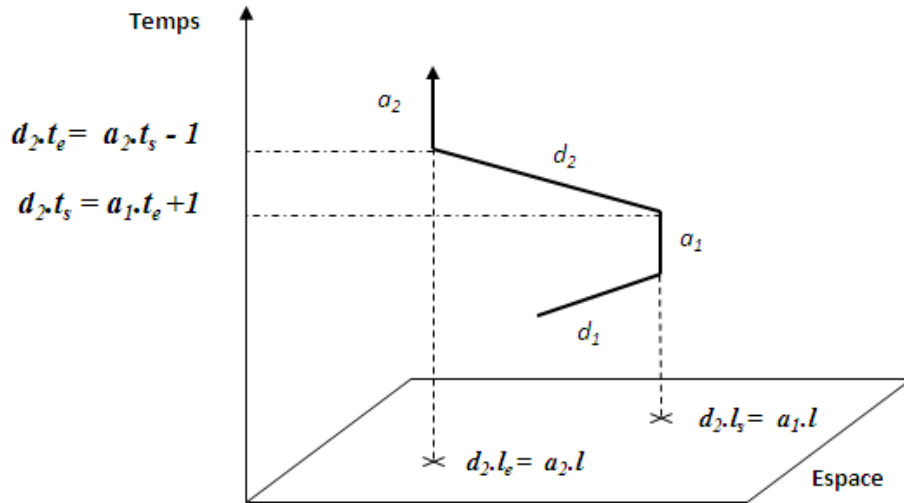
**Définition 1 (activité)** Une *activité* ( $a$ ) est une action qui se déroule à une certaine *localisation* ( $l$ ) de l'espace géographique pendant une période de temps non vide délimitée par un *temps de début* ( $t_s$ ) et un *temps de fin* ( $t_e$ ). Chaque activité possède un *type* d'activité qui décrit sémantiquement sa nature. Les types d'activités sont classés selon cette nature. On peut retrouver par exemple, s'il s'agit d'étudier le comportement de mobilité quotidienne de personnes, des activités de type "travailler", "faire des courses" ou "rester à la maison".

L'accomplissement des différentes activités nécessite un mouvement entre les localisations d'activités. Un tel mouvement est nommé *déplacement*.

#### 2.2.2 Déplacement

**Définition 2 (déplacement)** Un *déplacement* ( $d$ ) se déroule entre une *origine* ( $l_s$ ) et une destination ( $l_e$ ) spatiale. Comme pour une activité, un déplacement consomme une période de temps non vide. Il commence à un *temps de début* ( $t_s$ ) et se termine à un *temps de fin* ( $t_e$ ). Le *mode* est le moyen utilisé pour se déplacer. Il peut être inhérent à l'objet lui-même (marche à pied d'un individu) ou concerner un second objet mobile (e.g en voiture). Dans notre trajectoire, un unique déplacement peut s'effectuer selon plusieurs modes. Le déplacement entre une origine

et une destination suit un certain *itinéraire* spatial (*path*).



**Figure III.1** – Contraintes sur la trajectoire spatio-temporelle d'un objet mobile

### 2.2.3 Trajectoire spatio-temporelle

**Définition 3 (trajectoire spatio-temporelle)** La *trajectoire spatio-temporelle* (*tst*) d'un objet mobile, durant une période de temps donnée, est un cheminement d'événements qui sont soit des déplacements ( $d_i$ ), soit des activités ( $a_i$ ) (voir Figure III.1). Un objet mobile peut posséder une ou plusieurs trajectoires chacune définie sur une période de temps particulière. Le *but* d'un déplacement  $d_i$  est l'accomplissement de l'activité  $a_i$  qui suit (temporellement) immédiatement le déplacement. Nous supposons que le premier événement de la trajectoire est un déplacement tandis que le dernier est une activité, qu'il y a autant d'activités que de déplacements dans la trajectoire et que ce nombre est égal à  $n$ .

Les définitions présentées dans cette section, nous permettent d'atteindre un des objectifs de ce travail qui est de couvrir dans notre représentation les différentes dimensions spatiales, temporelles et sémantique de la trajectoire spatio-temporelle d'objets mobiles. D'autre part, d'un point de vue applicatif, il serait intéressant et utile pour l'utilisateur de la trajectoire de pouvoir gérer, manipuler et visualiser aisément et séparément en plus de l'aspect dynamique (c'est à dire le mouvement), un aspect statique (comme les arrêts) de la trajectoire qui n'est généralement pas pris en considération dans les représentations traditionnelles des trajectoires spatio-temporelles. Ces aspects sont parfaitement représentés, dans les définitions que nous proposons, par les notions de déplacements (qui représente le mouvement) et d'activités (qui représente les arrêts).

### 2.3 Contraintes sur la trajectoire spatio-temporelle

En examinant de près les définitions précédemment données et afin de garantir la représentation chaînée d'une trajectoire spatio-temporelle, nous constatons que nous devons dégager certaines contraintes de base qui vont nous servir lors de la spécification algébrique du type TST. Ces contraintes sont formulées par les équations (III.1).

#### 2.3.1 Contraintes spatiales

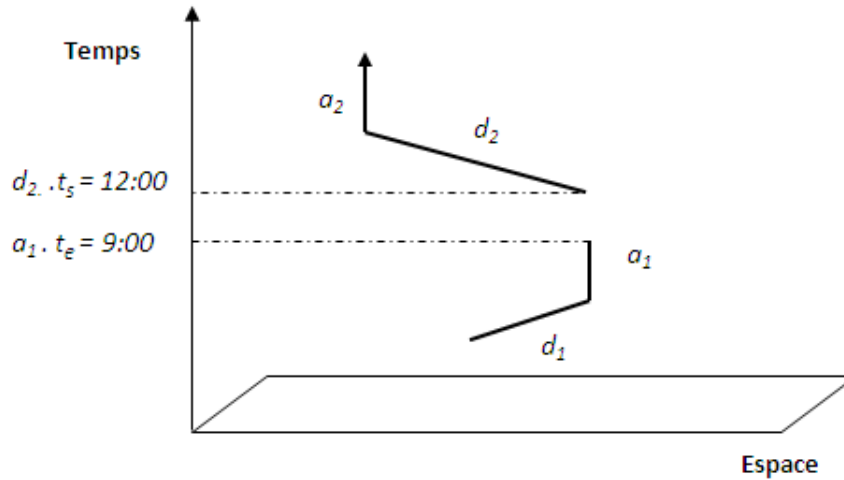
En effet, d'un point de vue spatial, le chaînage des activités et des déplacements au sein d'une trajectoire revient à préciser que l'origine (respectivement la destination) d'un déplacement est égal à la localisation de l'activité qui le précède (respectivement la suit) (voir Figure III.1). Ceci est traduit par les contraintes (1) et (2) dans le système d'équation (III.1).

#### 2.3.2 Contraintes temporelles

D'un point de vue temporel, le choix de la représentation de la succession temporelle des activités et des déplacements doit vérifier :

1. qu'il ne doit pas exister un instant temporel  $t$  pour lequel la trajectoire est définie mais tel qu'à cet instant l'objet n'est ni en mouvement, ni en activité. À titre d'exemple, dans la figure III.2, l'instant  $t = 10$  heures, pose un problème.
2. et que l'état de l'objet en mouvement doit être unique à tout instant. Autrement dit, à un instant  $t$  l'objet doit être soit en mouvement (donc en déplacement), soit au repos (donc en activité).

Pour ce faire, un choix doit être porté sur la nature des intervalles temporels des activités et des déplacements (fermés, ouverts ou semi-ouverts). Plusieurs cas de figure peuvent être envisagés, plus précisément 16 combinaisons sont possibles. Nous en retenons une, et nous décidons que pour ce travail, une activité,  $a$ , et un déplacement,  $d$ , sont définis temporellement comme suit :  $a=[t_s, t_e]$  et  $d=[t_s, t_e]$  et que les bornes supérieures et inférieures des intervalles temporels des activités et déplacements sont différents. Ceci est traduit par les contraintes (3) et (4) dans les équations (III.1). En effet, ces contraintes traduisent le fait le temps de début (temps de fin) d'un déplacement est égal au temps de fin (de début) de l'activité précédente (suivante) au quel on ajoute une unité de temps. L'unité de temps dépend de l'échelle temporelle choisie par l'utilisateur lors de la définition de la trajectoire spatio-temporelle (voir Figure III.1).



**Figure III.2** – Un cas de succession temporelle invalide dans la trajectoire spatio-temporelle

$$\begin{aligned}
 (1) \quad & d_i.l_s = a_{i-1}.l \quad \forall (1 < i \leq n) \\
 (2) \quad & d_i.l_e = a_i.l \quad \forall (1 \leq i \leq n) \\
 (3) \quad & d_i.t_s = a_{i-1}.t_e + 1 \quad \forall (1 < i \leq n) \\
 (4) \quad & d_i.t_e = a_i.t_s - 1 \quad \forall (1 \leq i \leq n)
 \end{aligned}$$

(III.1)

### 2.3.3 Contraintes de modélisation

Nous notons, enfin, que dans notre modélisation du comportement de déplacement qui se veut conforme au cadre de time geography, les cas de figure suivants ne seront pas pris en considération :

1. des entités qui ont deux ou plusieurs comportements similaires successifs (une activité suivie d'une activité ou un déplacement suivi par un déplacement),
2. des entités continuellement en mouvement comme les chauffeurs de bus ou de taxis,
3. des entités se déplaçant tout en exerçant des activités (e.g une personne qui se déplace en train et qui en même temps exerce des activités à bord de ce train).

Néanmoins, la flexibilité de ce modèle permet de s'adapter à une sémantique d'application un peu différente. En effet, la trajectoire d'un objet mobile s'interprète à partir de niveaux de

granularité spatiales et temporelles particuliers, et dans celui du contexte d'une classification d'activité particulière. Le choix pertinent de ces paramètres reste la tâche des utilisateurs. Ce choix détermine la sémantique du mouvement à modéliser et fixe ainsi quels cas de figure sont dégénérés et lesquels ne le sont pas. Par exemple, avec une granularité temporelle assez fine, les arrêts d'un chauffeur de taxi, qui sont généralement de courte durée, peuvent devenir perceptibles au sein de la trajectoire et révéler ainsi des informations pertinentes pour une application de surveillance de taxis, à un niveau de granularité plus général un tel déplacement exhibera une sémantique différente.

### 3 Besoins en analyse des trajectoires

Un modèle de trajectoire spatio-temporelle doit offrir des capacités opérationnelles et fonctionnelles permettant aux utilisateurs une analyse simple, efficace et couvrant de préférence les différentes composantes de la trajectoire. D'autre part, vu que la finalité de notre modèle est de créer un nouveau type de données destiné à être intégré dans un SGBD, le modèle doit proposer des mécanismes permettant d'assurer la cohérence de la base de données.

Dans cette section, nous essayons d'identifier les différents besoins en analyse de la trajectoire spatio-temporelle, telle que nous l'avons déjà définie dans la section précédente. Autrement dit, nous identifions différentes dimensions d'analyse et pour chaque dimension, nous dégagons, tout en gardant un certain niveau de généralité, différents sortes de manipulations de la trajectoire spatio-temporelle. Nous identifions, aussi, un ensemble de besoins qui servent à assurer la cohérence des bases de données manipulant des objets de type TST.

#### 3.1 Vérification de la cohérence de la base de données

Ce type de besoin est important lorsqu'il s'agit d'étendre un SGBD avec de nouvelles fonctionnalités. En effet, il est alors nécessaire de vérifier les contraintes d'intégrité des nouvelles données manipulées et ceci afin de maintenir la cohérence de la base de données. Dans le cas de la manipulation des trajectoires spatio-temporelles, les contraintes à vérifier sont en rapport avec la nature complexe et hybride de la donnée trajectoire.

#### 3.2 Analyse sémantique

Cette analyse concerne les aspects sémantiques de la trajectoire. Elle est inspirée, d'une part, de la nature de composition de la trajectoire qui est un chaînage entre activités et déplacements

et d'autre part de la nature des données sémantiques qui sont : comment se déplacer ? pourquoi se déplacer ? pourquoi s'arrêter ?. En effet, à une dimension sémantique, il peut exister différents besoins de manipulations de la trajectoires qui répondent à certains besoins d'analyses. Nous différencions, dans ce qui suit, différentes catégories de manipulations sémantiques de la trajectoire.

- **Comptage sémantique** : Vu que la conception de la trajectoire est un ensemble d'événements (activités et déplacements), il y aura probablement nécessité de connaître :
  - le nombre d'éléments de la trajectoire ; cette connaissance permettra de distinguer, par exemple, entre une trajectoire "chargée" en activités (et donc en déplacements) d'une autre moins "chargée" et ceci indépendamment de l'aspect temporel. Il est à noter, en effet, qu'une trajectoire sémantiquement "chargée" ne s'étale pas forcément sur une période de temps plus grande. À titre d'exemple, dans la Figure III.4, la trajectoire  $tst_1$  comporte plus d'éléments que la trajectoire  $tst_2$  mais s'étale sur une période de temps plus courte.
  - le nombre d'éléments vérifiant certains critères sémantiques comme le mode de déplacement, le but du déplacement ou la nature de l'arrêt. Cette dernière fonctionnalité peut donner une idée sur la fréquence d'exercice d'une activité ou d'un déplacement et révéler ainsi un comportement particulier de l'objet mobile (par exemple une fréquence élevée de l'activité de type *shopping* ou une utilisation trop fréquente de la *voiture* pour se déplacer).
- **Positionnement dans le TST** : Vu la composition chaînée de la trajectoire, des besoins de parcours et de positionnement des événements de la trajectoire se font ressentir. Ceci se traduit par des fonctionnalités :
  - de précedence/succession d'événements dans la trajectoire ; L'intérêt de ce genre de manipulation réside, essentiellement, dans la possibilité qu'elles doivent offrir à l'utilisateur du TST de parcourir les événements dans les deux sens : du début à la fin et vice versa. L'utilisateur peut ne s'intéresser qu'à un type d'événement et aura la possibilité de parcourir aisément le TST selon ce critère de choix.
  - et de renvoi de positions d'occurrence des événements dans la trajectoire.
- **Recherche sémantique dans le TST** : Ce genre de besoin opérationnel concerne la recherche d'activités ou de déplacements de la trajectoire selon différents critères sémantiques comme : leur position dans la trajectoire, leur mode de déplacement, leur objectif ou leur type.
- **Restriction sémantique** : S'il y'a besoin d'étudier une partie du comportement d'un objet en mouvement, on devrait être capable de restreindre sa trajectoire à une sous-

trajectoire. Cette sous-trajectoire sera composée d'un sous-ensemble d'activités et de déplacements. L'utilisateur de ce type d'opération devra être en mesure de spécifier les critères sémantiques de la restriction. Ces critères serviront à délimiter la trajectoire résultat.

- **Projection dans le domaine sémantique** : L'idée ici est d'effectuer une sorte de projection de la vision tri-dimensionnelle de la trajectoire, qui a été proposée dans le cadre de time geography, sur le "domaine" sémantique et d'en récupérer l'ensemble des activités ou des déplacements. Ce type d'opérations est utile dans le cas où l'utilisateur désire briser l'enchaînement et l'encapsulation dans le TST et récupérer les événements pour les étudier à part, sans avoir recours aux opérations du TST.
- **Similarité sémantique** : Il s'agit ici de tester si deux TST ont, dans le même ordre, eu des comportements sémantiques identiques. Autrement dit, si les deux TST ont le même nombre d'activités et que ces activités ont successivement les mêmes natures et ont été exécutées selon les mêmes modes de transport. Cette similarité est indépendante des aspects spatiaux et temporels, dans le sens où s'il s'avère que deux objets de type TST sont sémantiquement similaires ceci n'impose pas que leur parcours spatial soit similaire ou que les durées de leurs activités correspondantes soient égales (voir Figure III.3).

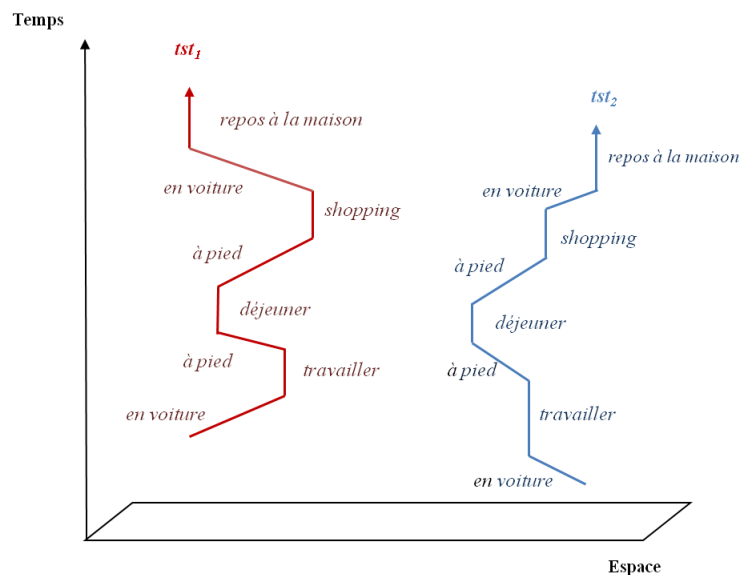
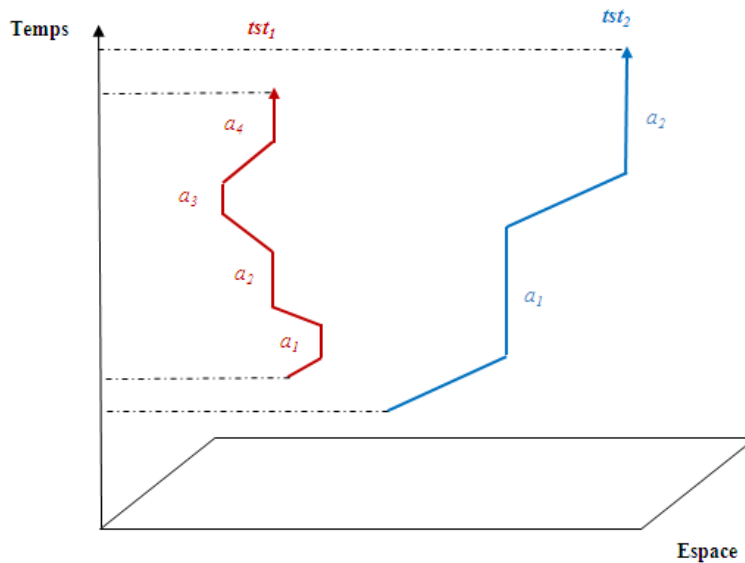


Figure III.3 – Exemple de deux trajectoires sémantiquement similaires

- **Prédicats** : Il s'agit ici des possibilités de tester si la trajectoire est vide ou si un événement donné appartient à la trajectoire.



**Figure III.4** – Exemple de trajectoires l'une chargée en activités ( $tst_1$ ) et l'autre moins chargée ( $tst_2$ ) et tel que  $tst_2$  dure une période de temps plus de longue

### 3.3 Analyse spatiale

Cette analyse concerne l'aspect purement spatial de la trajectoire. Nous dégagons, dans ce qui suit, différentes catégories de manipulations spatiales utiles pour la manipulation et donc l'analyse de la trajectoire spatio-temporelle.

- **Projection dans le domaine spatial** : Cette fonctionnalité répond au besoin de restreindre totalement ou partiellement la trajectoire à la dimension spatiale. Totalement, il s'agit de créer une "image spatiale" de la trajectoire; en fait c'est le *chemin spatial* ou *itinéraire* parcouru par l'objet mobile tout au long de la période de définition de la trajectoire. Partiellement, on pourrait s'intéresser aux localisations relatives aux lieux d'activités de l'objet en mouvement.
- **Relations spatiales** : Concerne essentiellement des relations de topologies spatiales entre trajectoires d'une part et entre trajectoires et entités spatiales représentant des éléments géographiques comme des points représentant des localisations, des polygones représentant des lacs ou des polygones représentant des routes, d'autre part.
- **Voisinage spatial** : Ce type de manipulation est relatif au besoin d'étude du voisinage spatial entre trajectoires d'un part, et entre trajectoires et entités spatiales comme des points, polygones, ou polygones d'autre part.
- **Calcul spatial** : Le calcul spatial nécessaire à la manipulation des trajectoires doit essentiellement être un calcul de :



- distance partielle ou totale parcourue par l’objet mobile au sein de cette trajectoire; cette dernière représente la taille spatiale de la trajectoire,
- distance entre trajectoires ou distance entre trajectoires et entités spatiales.
- **Recherche spatiale dans le TST** : Il s’agit ici d’effectuer une recherche d’éléments au sein de la trajectoire selon un critère spatial. Il peut s’agir, par exemple, de trouver les éléments de la trajectoires qui se sont déroulés à une localisation particulière (point spatial), selon une route particulière (un polygone), ou dans une région particulière (un polygone).
- **Similarité spatiale** : Il s’agit ici de tester si deux TST ont suivis le même parcours spatial.

### 3.4 Analyse temporelle

Cette analyse concerne la dimension purement temporelle de la trajectoire. Nous dégagons, dans ce qui suit, différentes catégories de manipulations spatiales utiles pour la manipulation et donc l’analyse de la trajectoire spatio-temporelle.

- **Relations temporelles** : Ce type de manipulation concerne les besoins de comparaisons à un niveau temporel des trajectoires et ceci à l’aide de l’application des relations temporelles entre deux trajectoires différentes d’une part et entre une trajectoire et une entité temporelle (un instant ou un intervalle temporel) d’autre part.
- **Calcul des durées** : Les objets mobiles consomment du temps durant leurs mouvements et même durant leurs arrêts. Il y a donc besoin de calculer :
  - la durée totale d’une trajectoire,
  - les durées totales des activités ou des déplacements d’une trajectoire.
- **Restriction temporelle** : Il s’agit ici de restreindre une trajectoire à un fragment de trajectoire étant donné un instant ou un intervalle de temps.
- **Projection dans le domaine temporel** : Cette fonctionnalité restreint la trajectoire à la dimension temporelle. Il s’agit de récupérer les instants ou les intervalles temporels de la trajectoire.
- **Recherche temporelle dans le TST** : Il s’agit ici d’effectuer une recherche d’éléments au sein de la trajectoire selon un critère temporel. Il peut s’agir, par exemple, de trouver les éléments de la trajectoires qui se sont déroulés à un instant de temps ou à un intervalle de temps particulier
- **Similarité temporelle** : Il s’agit ici de tester si deux TST se sont déroulés aux mêmes instants de temps. Autrement dit, deux TST sont dits similaires s’ils commencent au

même moment, et s'ils ont le même nombre d'activités et que ces activités commencent et se terminent, successivement, aux mêmes instants de temps.

### 3.5 Analyse spatio-temporelle

Les manipulations spatio-temporelles des trajectoires s'effectuent habituellement par des combinaisons de manipulations à la fois spatiales et temporelles. Nous estimons, qu'en plus de pouvoir effectuer ce genre de manipulation, il y a besoin d'offrir des mécanismes permettant une manipulation purement spatio-temporelle tels que :

- des opérations ensemblistes d'égalité, différence, union et intersection entre trajectoires,
- ou des opérations de positionnement ou de test de voisinage spatial à un instant de temps donné.

En s'appuyant sur cette étude sur les besoins de manipulation et d'analyse de la trajectoire spatio-temporelle, nous entamons dans la section suivante la spécification de notre type TST et nous proposons le modèle syntaxique qui représente la signature du TST.

## 4 Signature du TST

Le modèle syntaxique du type TST est composé de deux parties :

1. *Les sorts* : C'est l'ensemble des types de données nécessaires pour pouvoir définir le type abstrait de données TST.
2. *Les opérations* : Un type abstrait de données doit être manipulable à travers un ensemble bien défini d'opérations. Chaque opération est définie par sa *signature*. La signature d'une opération est composée du nom de l'opération, du produit cartésien des types de données des entrées et du type de données du résultat. La description d'une signature prend la forme suivante :

$$Type_1 \times Type_2 \times \dots \times Type_n \rightarrow Type_{Resultat}$$

Nous décrivons dans cette partie la signature des opérations sur le TST que nous avons identifiées. Ces opérations sont de différentes catégories et répondent aux besoins de manipulations de la trajectoire déjà recensés dans la section 3.

Nous présentons dans ce qui suit les détails de ces deux parties.

### 4.1 Sorts

Le recensement des besoins en analyse des trajectoires spatio-temporelles effectué dans la section 3 a révélé différentes dimensions d'analyse à savoir : une dimension spatiale, une dimension temporelle, une dimension sémantique et une dimension spatio-temporelle. À chaque dimension, nous avons dégagé certaines formes intéressantes de manipulation de la donnée trajectoire, comme le comptage d'activités/déplacements au sein d'une trajectoire ou la comparaison spatiale de deux trajectoires différentes. En s'appuyant sur cette étude, nous allons présenter les différents types de données dont l'intervention est nécessaire pour pouvoir définir le nouveau type TST.

En effet les *sorts* de la signature du nouveau type sont eux mêmes de différentes natures. Nous les recensons dans ce qui suit selon ce critère :

#### Les types atomiques et composés :

- Le type *Integer* : Le traitement de comptage au sein du TST produit des résultats qui doivent être des valeurs entières (e.g appartenant à l'ensemble  $\mathbb{N}$ ). Il y'a donc nécessité d'un type, appelé ici *Integer*, permettant d'instancier des valeurs entières.
- Le type *Real* : Le besoin de calcul de distances spatiales révèle le besoin d'un type, nommé ici *Real*, permettant d'instancier des valeurs réelles.
- Le type *Boolean* : Le besoin de comparaison entre trajectoires implique des traitements de tests d'existence de relations (spatiales, temporelles, etc.) entre celles ci. Ces traitements produisent généralement des résultats de nature booléenne, d'où le besoin du type *Boolean*.
- Le type *String* : Le caractère sémantique de la trajectoire se manifeste par des descriptions, généralement textuelle, comme le mode de déplacement "*voiture*" ou la nature d'activité "*shopping*". Nous avons donc besoin d'un type permettant d'instancier des valeurs textuelles, c'est ici le type *String*.
- Le type *Alist* : Certains besoins opérationnels associés au type TST, comme la recherche dans le TST, ont pour objectifs de retrouver un *ensemble* d'éléments (e.g activités ou déplacements) du TST. Nous avons donc besoin d'utiliser un type qui permet d'instancier un ensemble ordonné d'éléments de même type. C'est, ici, le type *Alist*.

**Les types spatiaux** : Les informations décrivant le TST sont en partie spatiales, comme par exemple la localisation où a lieu une activité. De plus, les besoins en comparaison spatiale des trajectoires a fait apparaître la nécessité de manipuler des entités spatiales, comme de tester le croisement entre la trajectoire d'une personne et une région ou une route.

- Le type *Point* : Ce type correspond à une entité ponctuelle à deux dimensions. À titre

d'exemple, une entité de type *Point* peut représenter dans le monde réel un lieu de travail ou de résidence.

- Le type *Polyline* : Ce type correspond à une entité linéaire dans le monde réel. Le type *Polyline* peut, par exemple, modéliser une route ou un lac.
- Le type *Polygon* : Ce type correspond à une entité polygonale (ou surfacique) du monde réel. Une valeur du type *Polygon* peut représenter une région d'intérêt, par exemple.

**Les types temporels :** Le besoin en types temporels dans la modélisation du type TST est évident. En effet, le caractère temporel de la trajectoire nécessite d'avoir recours aux primitives temporelles définie dans [23].

- Le type *Time* : Ce type correspond à un instant précis dans le temps.
- Le type *Interval* : Ce type correspond à un intervalle de temps fermé.

**Les types spatio-temporels/sémantiques :** Les types spatio-temporels *Activity* et *Trip* représentent l'activité et le déplacement d'une entité. En fait, selon les définitions données dans la section 2, une activité  $a$ , de type *Activity*, et un déplacement  $d$ , de type *Trip*, peuvent être définis formellement à l'aide de différents attributs spatiaux et temporels et dans certains cas spatio-temporels.

- *Activity* : En effet, dans le modèle que nous proposons, une activité  $a$  est une entité spatio-temporelle définie par le quadruple :

$$a = (l, t_s, t_e, type)$$

tels que l'attribut  $l \in Point$  est sa localisation, les attributs  $t_s$ , et  $t_e \in Time$  sont respectivement son temps de début et son temps de fin, l'attribut  $type \in String$  est son type. Ce dernier attribut permet de qualifier une activité selon un modèle descriptif qui peut être ajusté et enrichi en fonction du domaine d'application.

- *Trip* : Pareillement, un déplacement est formellement défini comme suit :

$$d = (l_s, l_e, t_s, t_e, mode, path)$$

tels que  $l_s$  et  $l_e \in Point$  représentent respectivement sa localisation d'origine et sa localisation de destination,  $t_s$  et  $t_e \in Time$  sont respectivement son temps de début et son temps de fin,  $mode \in Alist$  est la liste des modes de déplacement utilisés et *Path* représente son trajet spatial.

Selon le besoin applicatif, l'attribut *Path* peut représenter une simple information spatiale ou une information spatio-temporelle riche. Dans le premier cas, il s'agit d'une approche plutôt discrète du mouvement. les points d'intérêts essentiels sont les points de localisation des activités (e.g., l'application n'impose pas de connaître où se trouve l'entité mobile à un moment précis). Dans ce cas, l'attribut *Path* est le résultat de la projection du déplacement sur le plan spatial 2D et il peut simplement être représenté par une polyligne, c'est à dire

que  $path \in Polyline$ . Cet attribut peut alors être calculé à la demande, généré par une interpolation linéaire entre des points de localisations collectées au préalable, ou valué par des données issues de capteurs. Dans le second cas, il s'agit d'une approche continue du mouvement. Le trajet peut alors être en lui-même une sous-trajectoire exprimant le mouvement continu entre les localisations d'origine et de destination d'un déplacement. L'attribut *Path* peut alors être défini à l'aide du type spatio-temporel *mpoint* introduit dans [71], et on aura alors  $Path \in mpoint$ . Pour l'instant nous nous plaçons dans le contexte d'une approche discrète. En effet, analyser des positions successives, par exemple dans le cadre d'un déplacement, ne fait pas partie de nos objectifs mais pourrait être abordé par une approche concomitante à notre démarche. De même, notre modèle est extensible sans trop de difficultés pour couvrir une sémantique plus riche en changeant les types des attributs *type* et *mode*. Ce choix sera guidé principalement par les besoins applicatifs.

## 4.2 Opérations

Les potentialités du type TST sont illustrées par quelques opérations représentatives de son pouvoir d'expression et répondants aux besoins d'analyse recensés dans la section précédente. La signature des différentes catégories d'opérations sur le TST, accompagnée d'une description textuelle, sera donnée dans ce qui suit sous forme de tableaux.

### 4.2.1 Opérations de vérification de la cohérence de la base de données

Nous proposons un ensemble d'opérations permettant la vérification de l'intégrité des données de type TST. Ces opérations peuvent être utilisées par l'utilisateur lors de la création des données TST. Elles lui permettront ainsi d'éviter les erreurs en relation avec la structure interne complexe du type spatio-temporel et sémantiquement riche TST. Les opérations peuvent, aussi, être utilisées par d'autres opérations de manipulation du type afin d'assurer la cohérence de la base de données.

Pour ce faire, nous proposons, dans ce qui suit, quatre opérations de ce type. L'ensemble des signatures de chacune de ces opérations est illustré dans le tableau III.1. Il est à noter que ce type d'opérations a été proposé uniquement par [54] et dans le cadre spécifique d'une implémentation au sein du SGBD Oracle.

- l'opération *check\_activity* : cette opération vérifie qu'une activité donnée en argument est valide. La validité d'une activité peut être vérifiée sur deux plans : le plan temporel et le plan spatial. D'un point de vue temporel, une activité est valide si elle se déroule

durant une période de temps non vide. Sur un plan spatial, une activité est valide si sa localisation est bien du type géométrique spécifié par le modèle, qui est le type *Point* dans notre cas.

- l’opération *check\_trip* : cette opération vérifie qu’un déplacement donné en argument est valide. La validité d’un déplacement peut, aussi, être vérifiée sur deux plans : le plan temporel et le plan spatial. D’un point de vue temporel, un déplacement est valide s’il se déroule durant une période de temps non vide. Sur un plan spatial, un déplacement est valide si ses localisations de départ et d’arrivée sont bien de type *Point* et si le chemin parcouru par ce déplacement est bien de type *Polyline*.
- l’opération *check\_tst* : cette opération vérifie qu’une valeur de type TST donnée en argument est valide. Une donnée de type TST est valide si :
  - l’ensemble des activités et des déplacements qui la composent sont valides ;
  - et si l’ensemble des activités et des déplacements qui la composent vérifient les contraintes d’enchaînement spatio-temporel qui doit caractériser une donnée de type TST (voir équations (III.1)).
- l’opération *existe\_tst* : cette opération prend en argument une donnée *tst* de type TST et un instant temporel *t*. Elle vérifie, si à l’instant *t*, la valeur de *tst* est définie. Ceci évite aux utilisateurs du type TST de manipuler des valeurs non définies à certains instants temporels.

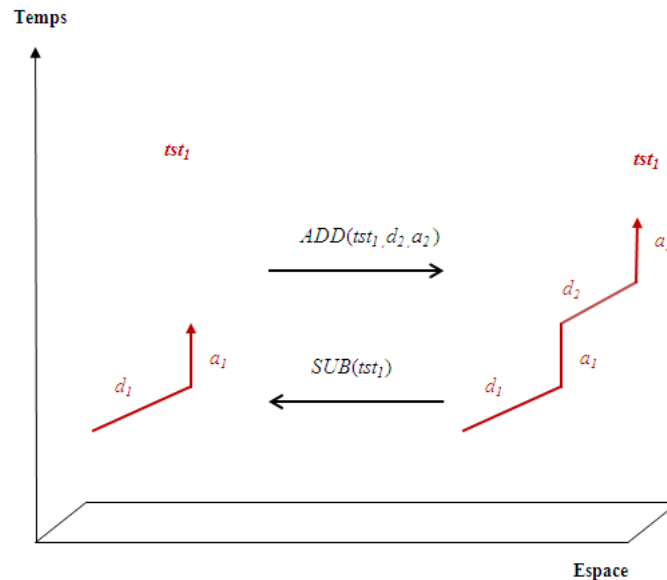
Opération	Signature	Description
<i>check_activity</i>	$Activity \rightarrow Boolean$ (III.2)	Vérifie la validité d’une activité
<i>check_trip</i>	$Trip \rightarrow Boolean$ (III.3)	Vérifie la validité d’un déplacement
<i>existe_tst</i>	$TST \times Instant \rightarrow Boolean$ (III.4)	Teste si, à un instant de temps, le TST existe
<i>check_tst</i>	$TST \rightarrow Boolean$ (III.5)	Vérifie la validité d’une valeur de type TST

**Tableau III.1** – Signature des opérations de vérification de la cohérence de la base de données

#### 4.2.2 Créateurs

Ce type d’opérations permet de créer ou de modifier un TAD. Nous distinguons pour le type TST trois créateurs qui sont *Make*, *Add* et *Sub*. L’initialisation d’une trajectoire est effectuée grâce à l’opération *Make* qui crée un TST vide. Le type TST est conçu comme un chaînage de déplacements et d’activités liés sémantiquement. Il ne peut exister un déplacement sans

objectif, qui est l'activité suivante. L'instanciation d'un TST se réalise par l'ajout d'un couple (déplacement, activité). De même, la suppression se fait en éliminant le dernier couple inséré (voir Figure III.5). Le tableau III.2 illustre la spécification de ces opérations.



**Figure III.5** – Exemple d'utilisation des opérations d'ajout et de suppression dans le TST

Opération	Signature	Description
Make	$\rightarrow TST$ (III.6)	Crée un TST vide
Add	$TST \times Trip \times Activity \rightarrow TST$ (III.7)	Ajoute à la fin d'un TST un déplacement suivi par une activité
Sub	$TST \rightarrow TST$ (III.8)	Enlève d'un TST la dernière activité ajoutée ainsi que le dernier déplacement

**Tableau III.2** – Signature des opérations de types créateurs de TST

### 4.2.3 Opérations sémantiques

Ce type d'opérations permet à l'utilisateur du type TST d'émettre des requêtes sur la sémantique de la trajectoire. Certaines de ces opérations, plus précisément les opérations de comptage sémantique ou celles de positionnement sémantique, sont inspirées de la liste des fonctionnalités temporelles définies pour la trajectoire spatio-temporelle dans [5]. Elles sont évidemment adaptée à notre modèle. Néanmoins, le reste des opérations sont nouvellement proposées en fonction

de l'analyse des besoins que nous avons effectuée dans la section précédente. La spécification des opérations que nous proposons est donnée dans ce qui suit.

**Opérations de comptage sémantique** Ces opérations calculent le nombre d'éléments de la trajectoire. Le calcul peut se faire en prenant ou pas en considération des critères sémantique. Elles sont de type numérique et retournent donc toutes un entier comme résultat. À titre d'exemple l'opération *Activity\_Count*, de signature (III.10), retourne le nombre d'activités du TST en argument alors que l'opération *Activities\_With\_Type\_Count*, de signature (III.12) retourne uniquement le nombre d'activités ayant comme nature la chaîne de caractère en argument. Le tableau III.3 donne la spécification des opérations de comptage sémantique que nous proposons.

Opération	Signature	Description
Trip_Count	$TST \rightarrow Integer$ (III.9)	Retourne le nombre de déplacements d'un TST
Activity_Count	$TST \rightarrow Integer$ (III.10)	Retourne le nombre d'activités d'un TST
Activities_With_Mode_Count	$TST \times String \rightarrow Integer$ (III.11)	Calcule le nombre d'activités du TST ayant été exercées en utilisant le mode indiqué
Activities_With_Type_Count	$TST \times String \rightarrow Integer$ (III.12)	Calcule le nombre d'activités du TST ayant le type indiqué
Trips_With_Purpose_Count	$TST \times String \rightarrow Integer$ (III.13)	Calcule le nombre de déplacements ayant pour objectif, l'objectif indiqué
Trips_With_Mode_Count	$TST \times String \rightarrow Integer$ (III.14)	Calcule le nombre de déplacements ayant pour mode, le mode indiqué

**Tableau III.3** – Signature des opérations de types comptage sémantique



**Opérations de positionnement** Ces opérations peuvent être utilisées afin d’avoir des informations concernant le positionnement d’un événement dans le chaînage déplacement/activité au sein d’un TST. Elles peuvent retourner :

1. un événement (une activité ou un déplacement) ayant une position particulière dans le chaînage ou vérifiant un critère de précédence ou de succession par rapport à un autre événement donné en argument. À titre d’exemple, l’opération *First\_Activity*, de signature (III.16), retourne la première activité d’un TST alors que l’opération *Activity\_Before\_Activity* (voir équation (III.27)) retourne l’activité qui précède une activité donnée en argument. La spécification de ces opérations de positionnement est donnée par le tableau III.4.
2. une liste d’événements vérifiant aussi un critère de précédence ou de succession par rapport à un autre événement donné en argument. À titre d’exemple l’opération *Activities\_Before\_Activity*, de signature (III.35), récupère l’ensemble des activités, précédant dans le chaînage du TST, une activité en argument. La spécification de ces opérations de positionnement est donnée par le tableau III.5.
3. un entier qui représente la position dans le chaînage d’une activité ou d’un déplacement donné, s’il existe bien sûr. À titre d’exemple, l’opération *Position\_Activity* va retourner la position de l’activité en argument dans le TST. Ainsi, étant donné une valeur *tst* de type TST, l’appel *Position\_Activity(tst, First\_Activity(tst))* va retourner l’entier 1 qui représente la position de la première activité de la trajectoire. La spécification des opérations de positionnement proposées est donnée dans le tableau III.6.

**Opérations de recherche sémantique** Ce type d’opérations permettent la recherche d’événements dans le TST selon des critères sémantiques, comme le mode de déplacement ou la nature de l’activité. Les opérations proposées retournent :

1. soit un événement vérifiant un critère sémantique donné en argument, comme l’opération *First\_Activity\_With\_Type*, de signature (III.39), qui cherche et retourne la première activité du TST ayant comme nature la chaîne de caractère donnée en argument. La signature de ce type d’opérations est donnée dans le tableau III.7.
2. soit une liste d’événements vérifiant un critère sémantique donné en argument. À titre d’exemple, l’opération *Trips\_With\_Purpose*, de signature (III.58), recherche et retourne la liste des déplacements ayant pour but la chaîne de caractère donnée en argument. La spécification de ces opérations est donnée dans le tableau III.8.

### III.4 Signature du TST

Opération	Signature	Description
First_Trip	$TST \rightarrow Trip$ (III.15)	Retourne le premier déplacement
First_Activity	$TST \rightarrow Activity$ (III.16)	Retourne la première activité
Last_Trip	$TST \rightarrow Trip$ (III.17)	Retourne le dernier déplacement
Last_Activity	$TST \rightarrow Activity$ (III.18)	Retourne la dernière activité
Nth_Trip	$TST \times Integer \rightarrow Trip$ (III.19)	Retourne le $n$ ème déplacement
Nth_Activity	$TST \times Integer \rightarrow Activity$ (III.20)	Retourne la $n$ ème activité
Activity_Before_Trip	$TST \times Trip \rightarrow Activity$ (III.21)	Retourne l'activité qui précède le déplacement en argument
Activity_After_Trip	$TST \times Trip \rightarrow Activity$ (III.22)	Retourne l'activité qui suit le déplacement en argument
Trip_Before_Activity	$TST \times Activity \rightarrow Trip$ (III.23)	Retourne le déplacement qui précède l'activité en argument
Trip_After_Activity	$TST \times Activity \rightarrow Trip$ (III.24)	Retourne le déplacement qui suit l'activité en argument
Trip_Before_Trip	$TST \times Trip \rightarrow Trip$ (III.25)	Retourne le déplacement qui précède le déplacement en argument
Trip_After_Trip	$TST \times Trip \rightarrow Trip$ (III.26)	Retourne le déplacement qui suit le déplacement en argument
Activity_Before_Activity	$TST \times Activity \rightarrow Activity$ (III.27)	Retourne l'activité qui précède l'activité en argument
Activity_After_Activity	$TST \times Activity \rightarrow Activity$ (III.28)	Retourne l'activité qui suit l'activité en argument

Tableau III.4 – Signature des opérations de positionnement sémantique retournant un évènement

## Chapitre III. Modèle syntaxique du TST

Opération	Signature	Description
Activities_Before_Trip	$TST \times Trip \rightarrow Alist$ (III.29)	Retourne toutes les activités qui précèdent le déplacement en argument
Activities_After_Trip	$TST \times Trip \rightarrow Alist$ (III.30)	Retourne toutes les activités qui suivent le déplacement en argument
Trips_Before_Activity	$TST \times Activity \rightarrow Alist$ (III.31)	Retourne tous les déplacements qui précèdent l'activité en argument
Trips_After_Activity	$TST \times Activity \rightarrow Alist$ (III.32)	Retourne tous les déplacements qui suivent l'activité en argument
Trips_Before_Trip	$TST \times Trip \rightarrow Alist$ (III.33)	Retourne tous les déplacements qui précèdent le déplacement en argument
Trips_After_Trip	$TST \times Trip \rightarrow Alist$ (III.34)	Retourne tous les déplacements qui suivent immédiatement le déplacement en argument
Activities_Before_Activity	$TST \times Activity \rightarrow Alist$ (III.35)	Retourne toutes les activités qui précèdent l'activité en argument
Activities_After_Activity	$TST \times Activity \rightarrow Alist$ (III.36)	Retourne toutes les activités qui suivent l'activité en argument

**Tableau III.5** – Signature des opérations de positionnement sémantique retournant une liste d'évènements

Opération	Signature	Description
Position_Activity	$TST \times Activity \rightarrow Integer$ (III.37)	Retourne la position d'occurrence d'une activité dans le chaînage
Position_Trip	$TST \times Trip \rightarrow Integer$ (III.38)	Retourne la position d'occurrence d'un déplacement dans le chaînage

**Tableau III.6** – Signature des opérations de positionnement sémantique retournant la position d'un événement dans le TST

Opération	Signature	Description
First_Activity_With_Type	$TST \times String \rightarrow Activity$ (III.39)	Retourne la première activité dont le type est celui indiqué
Last_Activity_With_Type	$TST \times String \rightarrow Activity$ (III.40)	Retourne la dernière activité dont le type est celui indiqué
Next_Activity_With_Type	$TST \times String \times Activity \rightarrow Activity$ (III.41)	Retourne la première activité, parmi celles qui suivent l'activité en argument, ayant pour type celui indiqué
Previous_Activity_With_Type	$TST \times String \times Activity \rightarrow Activity$ (III.42)	Retourne la première activité, parmi celles qui précèdent l'activité en argument, ayant pour type celui indiqué
First_Trip_With_Purpose	$TST \times String \rightarrow Trip$ (III.43)	Retourne le premier déplacement du TST dont l'objectif est celui indiqué
Last_Trip_With_Purpose	$TST \times String \rightarrow Trip$ (III.44)	Retourne le dernier déplacement du TST dont l'objectif est celui indiqué
Next_Trip_With_Purpose	$TST \times String \times Trip \rightarrow Trip$ (III.45)	Retourne le premier déplacement, parmi ceux qui suivent le déplacement en argument, ayant pour objectif celui indiqué
Previous_Trip_With_Purpose	$TST \times String \times Trip \rightarrow Trip$ (III.46)	Retourne le premier déplacement, parmi ceux qui précèdent le déplacement en argument, ayant pour objectif celui indiqué
First_Trip_With_Mode	$TST \times String \rightarrow Trip$ (III.47)	Retourne le premier déplacement du TST dont le mode est celui indiqué
Last_Trip_With_Mode	$TST \times String \rightarrow Trip$ (III.48)	Retourne le dernier déplacement du TST dont le mode est celui indiqué

Next_Trip_With_Mode	$TST \times String \times Trip \rightarrow Trip$ (III.49)	Retourne le premier déplacement, parmi ceux qui suivent le déplacement en argument, ayant pour mode de transport celui indiqué
Previous_Trip_With_Mode	$TST \times String \times Trip \rightarrow Trip$ (III.50)	Retourne le premier déplacement, parmi ceux qui précèdent le déplacement en argument, ayant pour mode de transport celui indiqué
First_Activity_With_Mode	$TST \times String \rightarrow Activity$ (III.51)	Retourne la première activité du TST, ayant pour mode celui indiqué
Last_Activity_With_Mode	$TST \times String \rightarrow Activity$ (III.52)	Retourne la dernière activité du TST, ayant pour mode celui indiqué
Next_Activity_With_Mode	$TST \times String \times Activity \rightarrow Activity$ (III.53)	Retourne la première activité, parmi celles qui suivent l'activité en argument, ayant pour mode celui indiqué
Previous_Activity_With_Mode	$TST \times String \times Activity \rightarrow Activity$ (III.54)	Retourne la première activité, parmi celles qui précèdent l'activité en argument, ayant pour mode celui indiqué

**Tableau III.7** – Signature des opérations de recherche sémantique retournant un événement

### III.4 Signature du TST

Opération	Signature	Description
Activities_With_Mode	$TST \times String \rightarrow Alist$ (III.55)	Retourne toutes les activités dont le mode de transport est celui indiqué
Activities_With_Type	$TST \times String \rightarrow Alist$ (III.56)	Retourne toutes les activités dont le type est celui indiqué
Trips_With_Mode	$TST \times String \rightarrow Alist$ (III.57)	Retourne tous les déplacements dont le mode est celui indiqué
Trips_With_Purpose	$TST \times String \rightarrow Alist$ (III.58)	Retourne tous les déplacements dont l'objectif est celui indiqué

**Tableau III.8** – Signature des opérations de recherche sémantique retournant une liste d'événements

**Opérations de restriction sémantique** Une opération de restriction sémantique crée une nouvelle trajectoire composée d'un sous-ensemble de la trajectoire en entrée. Ces nouvelles "sous-trajectoires" sont créées selon des arguments sémantiques donnés en entrée et qui sont soit des activités, soit des déplacements. Le rôle de ces arguments est de délimiter le fragment de trajectoire à retenir dans la trajectoire initiale. La spécification de la totalité des opérations de restriction sémantique est donnée dans le tableau III.9. À titre d'exemple, nous proposons l'opération *At\_Activity\_Activity* (voir la signature (III.59)).

Opération	Signature	Description
At_Activity_Activity	$TST \times Activity \times Activity \rightarrow TST$ (III.59)	Restreint sémantiquement le TST selon deux activités
At_Trip_Trip	$TST \times Trip \times Trip \rightarrow TST$ (III.60)	Restreint sémantiquement le TST selon deux déplacements

**Tableau III.9** – Signature des opérations de restriction sémantique

Cette opération donne en résultat une trajectoire "incluse" dans la trajectoire en argument. La trajectoire résultat doit avoir comme première et dernière activités celles qui sont en arguments. Dans la Figure III.6, la trajectoire  $tst_2$  est le résultat de l'opération  $At\_Activity\_Activity(tst_1, a_1, a_3)$ . Cette opération est particulièrement utile pour restreindre une trajectoire à un seul couple activité/déplacement. Dans ce cas, l'utilisateur aura accès aux événements du TST, toujours sous la forme d'une unique entité TST, et pourra de telle sorte

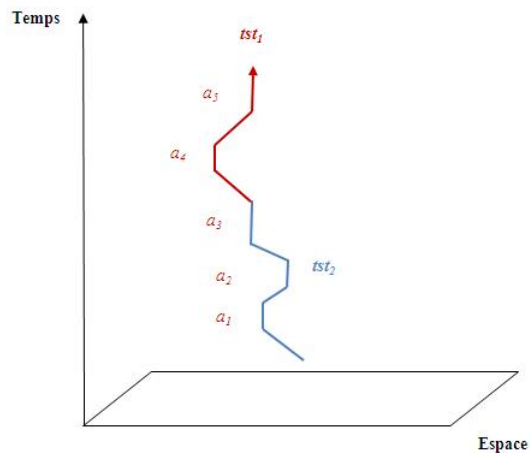


Figure III.6 – Un exemple de restriction sémantique d’un TST

étudier chaque couple d’événements individuellement en appliquant différentes sortes d’opérateurs sur les nouveaux TST créés.

**Opérations de projection dans le domaine sémantique** Nous proposons deux opérations de projection sémantique. Ces opérations prennent en argument un TST et retournent tout simplement une valeur de type *Alist* qui représente la liste de ses activités ou de ses déplacements. La spécification de ces opérations est donnée dans le tableau III.10.

Opération	Signature	Description
Activity_List	$TST \rightarrow Alist(III.61)$	Retourne la liste des activités d’un TST
Trip_List	$TST \rightarrow Alist(III.62)$	Retourne la liste des déplacements d’un TST

Tableau III.10 – Signature des opérations de projection dans le domaine sémantique

**Opérations de similarité sémantique** Nous proposons une unique opération de test de similarité sémantique. Cette opération prend en argument deux TST et retourne un booléen qui indique l’existence ou pas d’une telle similarité entre les deux TST en arguments. La spécification de cette opération est illustrée dans le tableau III.11.

**Prédicats** Toutes ces opérations retournent des valeurs booléennes. Certaines, comme l’opération *Include\_Activity*, de signature (III.64), testent l’appartenance d’une activité ou d’un déplacement au TST. Une autre opération, nommée *Empty*, de signature (III.66), teste si le

Opération	Signature	Description
Semantic_Similarity	$TST \times TST \rightarrow Boolean$ (III.63)	Teste si deux TST donnés sont sémantiquement similaires

**Tableau III.11** – Signature des opérations de similarité sémantique

TST est vide. Un TST est considéré vide s'il ne contient aucun couple activité/déplacement. La spécification de ces opérations est donnée dans le tableau III.12

Opération	Signature	Description
Include_Activity	$TST \times Activity \rightarrow Boolean$ (III.64)	Teste si une activité donnée appartient à un TST
Include_Trip	$TST \times Trip \rightarrow Boolean$ (III.65)	Teste si un déplacement donné appartient à un TST
Empty	$TST \rightarrow Boolean$ (III.66)	Teste si le TST est vide

**Tableau III.12** – Signature des opérations de types prédicats sémantiques

#### 4.2.4 Opérations spatiales

Ce type d'opérations permet à l'utilisateur du type TST d'émettre des requêtes sur l'aspect purement spatial de la trajectoire. Dans ce sens, nous avons adapté certaines sous-catégories d'opérations proposées dans [71], comme celles de projection ou de restriction dans le domaine spatial. Nous avons en plus proposé d'autres opérations répondant aux besoins d'analyses des trajectoires comme les opérations de recherche spatiale, celles de relations spatiales ou comme les relations de similarité spatiale. La spécification des opérations que nous proposons est donnée dans ce qui suit.

**Opérations de projection dans le domaine spatial** Ce type d'opérations retournent des entités spatiales comme des polygones, des points ou des listes de points qui représentent la projection spatiale totale ou partielle du TST. À titre d'exemple, nous proposons l'opération *Itinerary*, de signature (III.67), qui retourne un polygone représentant le chemin spatial parcouru par une trajectoire donnée en argument. Dans la figure III.7, cette opération appliquée à la trajectoire *tst* retourne le polygone *l*. Une autre opération appelée *Points* (voir signature (III.69)) retourne la liste des points spatiaux correspondants aux lieux activités de la trajec-



## Chapitre III. Modèle syntaxique du TST

toire. Dans la figure III.7, ces lieux correspondent aux points  $p_1$ ,  $p_2$  et  $p_3$ . La spécification de ces opérations est donnée dans le tableau III.13.

Opération	Signature	Description
Itinerary	$TST \rightarrow Polyline$ (III.67)	Retourne l'itinéraire spatial d'un TST
Path_Between	$TST \times Activity \times Activity \rightarrow Polyline$ (III.68)	Retourne le chemin spatial parcouru entre deux activités
Points	$TST \rightarrow Alist$ (III.69)	Retourne tous les points d'intérêts d'un TST
First_Point	$TST \rightarrow Point$ (III.70)	Retourne le point spatial de départ du TST
End_Point	$TST \rightarrow Point$ (III.71)	Retourne le point spatial de fin du TST

**Tableau III.13** – Signature des opérations de projection dans le domaine spatial

**Opérations de recherche spatiale** Ces opérations prennent en arguments un TST et une entité spatiale (point, polyligne ou région) et retournent une activité, un déplacement ou une liste d'activités ou de déplacements qui vérifient une relation spatiale particulière avec l'entité spatiale en argument. La spécification et l'explication de ces opérations sont données dans le tableau III.14.

Opération	Signature	Description
Activities_At_Point	$TST \times Point \rightarrow Alist$ (III.72)	Retourne les activités qui se sont déroulées à un point donné
FirstActivity_At_Point	$TST \times Point \rightarrow Activity$ (III.73)	Retourne la première activité qui a eu lieu à un point donné
LastActivity_At_Point	$TST \times Point \rightarrow Activity$ (III.74)	Retourne la dernière activité qui a eu lieu à un point donné

### III.4 Signature du TST

Trips_At_Point	$TST \times Point \rightarrow Alist$ (III.75)	Retourne les déplacements qui sont passés par un point donné
FirstTrip_At_Point	$TST \times Point \rightarrow Trip$ (III.76)	Retourne le premier déplacement du TST qui est passé par un point donné
LastTrip_At_Point	$TST \times Point \rightarrow Trip$ (III.77)	Retourne le dernier déplacement du TST qui a est passé par un point donné
Activities_At_Polyline	$TST \times Polyline \rightarrow Alist$ (III.78)	Retourne les activités qui ont eu lieu à une ligne donnée
FirstActivity_At_Polyline	$TST \times Polyline \rightarrow Activity$ (III.79)	Retourne la première activité qui a eu lieu à une ligne donnée
LastActivity_At_Polyline	$TST \times Polyline \rightarrow Activity$ (III.80)	Retourne la dernière activité qui a eu lieu à une ligne donnée
Trips_At_Polyline	$TST \times Polyline \rightarrow Alist$ (III.81)	Retourne les déplacements qui se sont déroulés selon une ligne donnée
FirstTrip_At_Polyline	$TST \times Polyline \rightarrow Trip$ (III.82)	Retourne le premier déplacement qui s'est déroulé selon une ligne donnée
LastTrip_At_Polyline	$TST \times Polyline \rightarrow Trip$ (III.83)	Retourne le dernier déplacement qui s'est déroulé selon une ligne donnée
Activities_At_Region	$TST \times Region \rightarrow Alist$ (III.84)	Retourne les activités qui se sont déroulées à l'intérieur d'une région donnée
FirstActivity_At_Region	$TST \times Region \rightarrow Activity$ (III.85)	Retourne la première activité qui s'est déroulée à l'intérieur d'une région donnée

LastActivity_At_Region	$TST \times Region \rightarrow Activity$ (III.86)	Retourne la dernière activité qui s'est déroulée à l'intérieur d'une région donnée
Trips_At_Region	$TST \times Region \rightarrow Alist$ (III.87)	Retourne les déplacements qui se sont déroulés à une région donnée
FirstTrip_At_Region	$TST \times Region \rightarrow Trip$ (III.88)	Retourne le premier déplacement qui s'est déroulé à une région donnée
LastTrip_At_Region	$TST \times Region \rightarrow Trip$ (III.89)	Retourne le dernier déplacement qui s'est déroulé à une région donnée

Tableau III.14 – Signature des opérations de recherche spatiale

**Opérations de relations spatiales** Les relations spatiales dites topologiques sont définies à partir de primitives géométriques dans un espace à deux dimensions [1, 21]. On retrouve ainsi, par exemple, des relations topologiques entre des couples *Polyline/Point*, *Polyline/Polyline*, *Polyline /Polygon*. Sur la base de ces relations, nous avons spécifié un ensemble de relations de type : *TST/Point* (par analogie aux relations *Polyline/Point*), *TST/Polyline* (*Polyline/Polyline*), *TST/Polygon* (*Polyline/Polygon*) et *TST/TST* (*Polyline/Polyline*). La spécification de ces opérations est donnée dans le tableau III.15.

Opération	Signature	Description
<b>TST/Point</b>		
TST_Contains_Point	$TST \times Point \rightarrow Boolean$ (III.90)	Teste si un TST contient spatialement un point
TST_EndsBy_Point	$TST \times Point \rightarrow Boolean$ (III.91)	Teste si un TST se termine spatialement par un point
TST_BeginsBy_Point	$TST \times Point \rightarrow Boolean$ (III.92)	Teste si un TST commence spatialement par un point
<b>TST/Polyline</b>		

### III.4 Signature du TST

TST_Equals_Polyline	$TST \times Polyline \rightarrow Boolean$ (III.93)	Teste si un TST est spatialement égal à une poly-ligne
TST_Cross_Polyline	$TST \times Polyline \rightarrow Boolean$ (III.94)	Teste si un TST croise spatialement une poly-ligne
TST_Disjoint_Polyline	$TST \times Polyline \rightarrow Boolean$ (III.95)	Teste si un TST et une poly-ligne sont spatialement disjoints
TST_Contains_Polyline	$TST \times Polyline \rightarrow Boolean$ (III.96)	Teste si un TST contient spatialement une poly-ligne
<b>TST/Polygon</b>		
TST_Inside_Region	$TST \times Polygon \rightarrow Boolean$ (III.97)	Teste si un TST est spatialement à l'intérieur d'un polygone
TST_Touch_Region	$TST \times Polygon \rightarrow Boolean$ (III.98)	Teste si un TST touche spatialement la frontière d'un polygone
TST_Cross_Region	$TST \times Polygon \rightarrow Boolean$ (III.99)	Teste si un TST croise spatialement une région
TST_Disjoint_Region	$TST \times Polygon \rightarrow Boolean$ (III.100)	Teste si un TST et un polygone sont spatialement disjoints
<b>TST/TST</b>		
TST_SpatialEqual_TST	$TST \times TST \rightarrow Boolean$ (III.101)	Teste si deux TST sont spatialement égaux
TST_SpatialDisjoint_TST	$TST \times TST \rightarrow Boolean$ (III.102)	Teste si deux TST sont spatialement disjoints

TST_SpatialCross_TST	$TST \times TST \rightarrow Boolean$ (III.103)	Teste si deux TST se croisent dans l'espace
TST_SpatialContains_TST	$TST \times TST \rightarrow Boolean$ (III.104)	Teste si un TST contient spatialement un autre TST

**Tableau III.15** – Signature des opérations de relations spatiales sur un TST

**Opérations de voisinage spatial** Ces opérations servent à tester la proximité spatiale, selon une distance donnée, entre deux TST, entre un TST et un point, entre un TST et un polygone et entre un TST et une région. La spécification de ces opérations est donnée dans le tableau [III.16](#).

**Opérations de calcul spatial** Ces opérations retournent des valeurs numériques (un réel pour la plupart) représentant des distances parcourues au sein de la trajectoire. La taille spatiale d'une trajectoire peut, à titre d'exemple, être calculée par l'opération *Spatial\_Length*, de signature ([III.109](#)), qui retourne la distance totale parcourue par l'objet en mouvement. La distance partielle parcourue pour accéder d'une activité à une autre activité peut être calculée par l'opération *Length\_Path\_Between*, de signature ([III.110](#)). Nous avons aussi définis des opérations de calcul de distances entre deux objets de type TST et entre un TST et une entité spatiale. La spécification de ces opérations est donnée dans le tableau [III.17](#).

**Opérations de similarité spatiale** Nous proposons une unique opération de similarité spatiale. Cette opération prend en argument deux TST et retourne un booléen qui indique si ces deux TST ont suivis, tout au long de leur période de définition, exactement le même chemin spatial. La spécification de cette opération est donnée dans le tableau [III.18](#).

### 4.2.5 Opérations temporelles

Comme pour les autres catégories d'opérations, nous proposons un ensemble d'opérations de manipulation de trajectoires spatio-temporelle à un niveau temporel. Certaines propositions ont été inspirées et adaptées des travaux de [71], comme les opérations de projection ou de restriction temporelle. De plus, nous proposons, dans cette catégories de manipulation, d'autres opérations temporelles comme les opérations de relations temporelles, qui n'ont été proposées dans [71] que pour les type dit "non temporel", et les opérations de similarité temporelle.

### III.4 Signature du TST

Opération	Signature	Description
TST_SpatialNeighbour_TST	$TST \times TST \times Real \rightarrow Boolean$ (III.105)	Teste si deux TST sont spatialement voisins selon une distance de voisinage donnée
TST_SpatialNeighbour_Point	$TST \times Point \times Real \rightarrow Boolean$ (III.106)	Teste si un TST et un point sont spatialement voisins selon une distance de voisinage donnée
TST_SpatialNeighbour_Polyline	$TST \times Polyline \times Real \rightarrow Boolean$ (III.107)	Teste si un TST et une polyligne sont spatialement voisins selon une distance de voisinage donnée
TST_SpatialNeighbour_Region	$TST \times Region \times Real \rightarrow Boolean$ (III.108)	Teste si un TST et une région sont spatialement voisins selon une distance de voisinage donnée

**Tableau III.16** – Signature des opérations de voisinage spatial

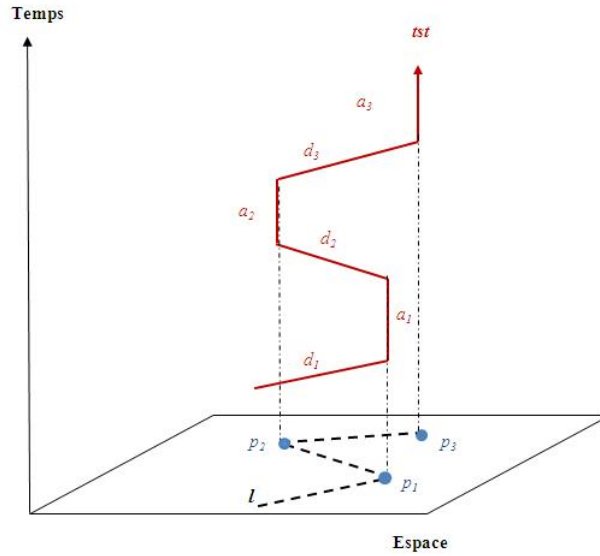


Figure III.7 – Exemple de projections spatiales du TST

Opération	Signature	Description
Spatial_Length	$TST \rightarrow Real$ (III.109)	Retourne la taille spatiale d'un TST
Length_Path_Between	$TST \times Activity \times Activity \rightarrow Real$ (III.110)	Retourne la distance parcourue entre deux activités d'un TST
TST_Distance_TST	$TST \times TST \rightarrow Real$ (III.111)	Calcule la distance spatiale entre deux TST
TST_Distance_Point	$TST \times Point \rightarrow Real$ (III.112)	Calcule la distance spatiale entre un TST et un point
TST_Distance_Polyline	$TST \times Polyline \rightarrow Real$ (III.113)	Calcule la distance spatiale entre un TST et une ligne
TST_Distance_Region	$TST \times Region \rightarrow Real$ (III.114)	Calcule la distance spatiale entre un TST et une région

Tableau III.17 – Signature des opérations de calcul spatial

Opération	Signature	Description
Spatial_Similarity	$TST \times TST \rightarrow Boolean$ (III.115)	Teste si deux TST sont spatialement similaires

Tableau III.18 – Signature de l'opération de similarité spatiale

**Opérations de relations temporelles** Ce type d'opérations concerne les relations temporelles définies à partir des algèbres de manipulation des types temporels introduite dans [23, 34]. Ces algèbres permettent de manipuler les types *Time* et *Interval*. Les différentes relations temporelles entre ces types permettent de définir pour le type TST un ensemble d'opérations manipulant des couples *Time/TST* (*Time/Interval*), *Interval/TST* (*Interval/Interval*) et *TST/TST* (*Interval/Interval*).

La sémantique et l'application de ces opérations sont illustrées par l'exemple de la Figure III.8 avec les entités suivantes : considérons les trajectoires *tst1* et *tst2*, une activité *a*, un déplacement *d*, un point temporel *t*, et un intervalle *I* de bornes *t<sub>1</sub>* et *t<sub>2</sub>*. L'exemple présenté donne les relations suivantes : "*t<sub>1</sub>* *During a*", "*t<sub>1</sub>* *Before d*", "*t* *Ends tst<sub>2</sub>*", "*I* *During tst<sub>1</sub>*" et "*tst<sub>1</sub>* *After tst<sub>2</sub>*". La spécification des opérations de relations temporelles est donnée dans le tableau III.19.

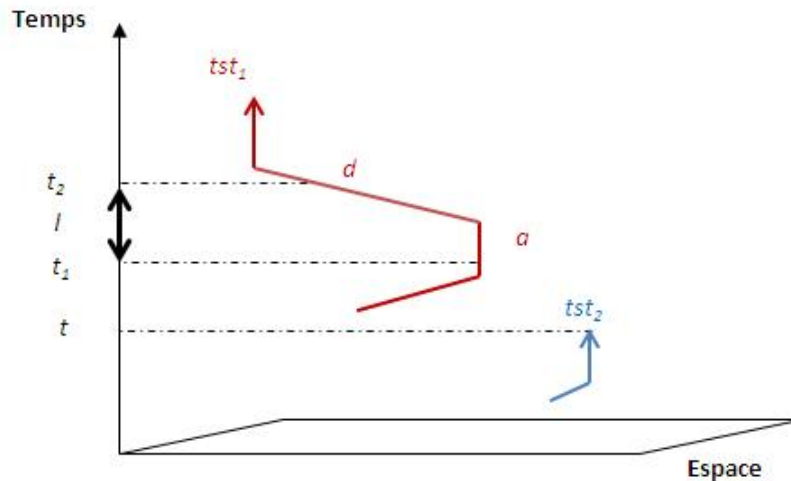


Figure III.8 – Exemple de relations temporelles



Opération	Signature	Description
<b>TST/Time</b>		
TST_Before_Time	$TST \times Time \rightarrow Boolean$ (III.116)	Teste si un TST précède un instant de temps
TST_After_Time	$TST \times Time \rightarrow Boolean$ (III.117)	Teste si un TST succède à un instant de temps
TST_EndedBy_Time	$TST \times Time \rightarrow Boolean$ (III.118)	Teste si un TST se termine par un instant de temps
TST_Contains_Time	$TST \times Time \rightarrow Boolean$ (III.119)	Teste si un TST contient un instant de temps
TST_BegunBy_Time	$TST \times Time \rightarrow Boolean$ (III.120)	Teste si un TST commence par un instant de temps
<b>TST/Interval</b>		
TST_Before_Interval	$TST \times Interval \rightarrow Boolean$ (III.121)	Teste si un TST précède un intervalle de temps
TST_After_Interval	$TST \times Interval \rightarrow Boolean$ (III.122)	Teste si un TST succède à un intervalle de temps
TST_Meets_Interval	$TST \times Interval \rightarrow Boolean$ (III.123)	Teste si un TST rencontre un intervalle de temps
TST_Overlaps_Interval	$TST \times Interval \rightarrow Boolean$ (III.124)	Teste un TST couvre un intervalle de temps
TST_Begins_Interval	$TST \times Interval \rightarrow Boolean$ (III.125)	Teste si un TST commence un intervalle de temps
TST_BegunBy_Interval	$TST \times Interval \rightarrow Boolean$ (III.126)	Teste si un TST commence par un intervalle de temps

TST_During_Interval	$TST \times Interval \rightarrow Boolean$ (III.127)	Teste si un TST se déroule pendant un intervalle de temps
TST_Equals_Interval	$TST \times Interval \rightarrow Boolean$ (III.128)	Teste si un TST est égal à un intervalle de temps
TST_OverlappedBy_Interval	$TST \times Interval \rightarrow Boolean$ (III.129)	Teste si un TST est couvert par un intervalle de temps
TST_Ends_Interval	$TST \times Interval \rightarrow Boolean$ (III.130)	Teste si un TST termine un intervalle de temps
TST_EndedBy_Interval	$TST \times Interval \rightarrow Boolean$ (III.131)	Teste si un TST se termine par un intervalle de temps
TST_MetBy_Interval	$TST \times Interval \rightarrow Boolean$ (III.132)	Teste si un TST rencontre un intervalle de temps
<b>TST/TST</b>		
TST_Before_TST	$TST \times TST \rightarrow$ (III.133)	Teste si un TST précède un autre TST
TST_After_TST	$TST \times TST \rightarrow Boolean$ (III.134)	Teste si un TST succède à un autre TST
TST_Meets_TST	$TST \times TST \rightarrow Boolean$ (III.135)	Teste si un TST rencontre un autre TST
TST_Overlaps_TST	$TST \times TST \rightarrow Boolean$ (III.136)	Teste si un TST couvre un autre TST
TST_Begins_TST	$TST \times TST \rightarrow Boolean$ (III.137)	Teste si un TST commence un autre TST
TST_BegunBy_TST	$TST \times TST \rightarrow Boolean$ (III.138)	Teste si un TST est commencé par un autre TST
TST_During_TST	$TST \times TST \rightarrow Boolean$ (III.139)	Teste si un TST se déroule pendant un intervalle de temps

TST_TemporalEquals_TST	$TST \times TST \rightarrow Boolean$ (III.140)	Teste si un TST est égal à un autre TST
TST_OverlappedBy_TST	$TST \times TST \rightarrow Boolean$ (III.141)	Teste si un TST est couvert par un autre TST
TST_Ends_TST	$TST \times TST \rightarrow Boolean$ (III.142)	Teste si un TST termine un autre TST
TST_EndedBy_TST	$TST \times TST \rightarrow Boolean$ (III.143)	Teste si un TST est terminé par un autre TST
TST_MetBy_TST	$TST \times TST \rightarrow Boolean$ (III.144)	Teste si un TST rencontre un autre TST

**Tableau III.19** – Signature des opérations de relations temporelles

**Opérations de calcul des durées** Ces opérations prennent en argument un TST et retournent un réel qui indique une durée. Cette durée peut être celle de la période de définition du TST, celle de la période totale d'exercice des activités de l'objet mobile dans le TST ou celle de la période totale des déplacements de l'objet dans le TST. La spécification des opérations de relations temporelles est donnée dans le tableau III.20.

Opération	Signature	Description
Duration	$TST \rightarrow Real$ (III.145)	Retourne la durée temporelle du TST
Activities_Duration	$TST \rightarrow Real$ (III.146)	Retourne la durée totale de toutes les activités du TST
Trips_Duration	$TST \rightarrow Real$ (III.147)	Retourne la durée totale de tous les déplacements du TST

**Tableau III.20** – Signature des opérations de calcul de durées temporelles

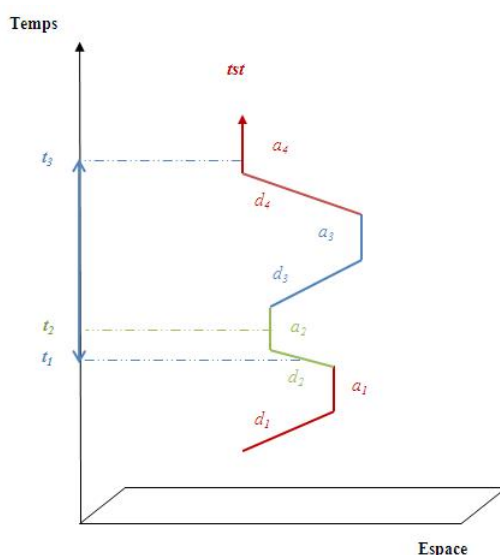
**Opérations de restriction temporelle** Comme pour les opérations de restriction spatiale et sémantique, les opérations de restriction temporelle retournent un nouveau TST qui est en fait la sélection d'une partie du TST donné en argument. Cette sélection se fait selon des arguments temporels. Nous proposons, en tout, deux opérations de ce type. La spécification des opérations de restriction temporelle est donnée dans le tableau III.21.

Opération	Signature	Description
At_Time	$TST \times Time \rightarrow TST$ (III.148)	Restreint le TST à un temps donné
At_Interval	$TST \times Interval \rightarrow TST$ (III.149)	Restreint le TST à un intervalle donné

**Tableau III.21** – Signature des opérations de restriction temporelle

Étant donné un instant de temps  $t$ , l'opération  $At\_Time$  restreint le TST à un unique couple (déplacement, activité), si l'un des éléments de ce couple s'est déroulé à l'instant  $t$ . Ceci indique sémantiquement qu'à l'instant  $t$  l'objet en mouvement est soit en déplacement, soit en activité. À titre d'exemple dans la Figure III.9, à l'instant  $t_2$ , l'opération  $At\_time$  appliquée à la trajectoire  $tst$  retourne le TST composé du couple  $(d_2, a_2)$ . La deuxième opération proposée est l'opération  $At\_Interval$ . Étant donné un intervalle de temps, cette opération restreint le TST

en argument au TST complètement temporellement incluse dans l'intervalle en question. C'est à dire que sémantiquement, le TST de retour illustre les couples d'activités et de déplacements du TST en argument qui se sont déroulés pendant l'intervalle en question. Il se peut qu'il y'ait un déplacement qui vérifie cette condition mais si l'activité qui le précède ne la vérifie pas, ni l'un ni l'autre ne sera retenu dans la sélection de la restriction temporelle. À titre d'exemple dans la Figure III.9, l'opération *At\_Interval* appliquée à la trajectoire *tst* et l'intervalle de temps délimité par les instants  $t_1$  et  $t_3$  retournera un nouveau TST composé uniquement du couple  $(d_3, a_3)$ .



**Figure III.9** – Exemple d'opérations de restriction temporelle

**Opérations de projection dans le domaine temporel** Ces opérations projettent le TST dans le domaine temporel en retournant des instant de temps, des intervalles ou une liste d'entités temporelles. Nous proposons à titre d'exemple l'opération *Temporal\_Length* :  $TST \rightarrow Real$  qui prend un TST en argument et retourne l'intervalle de définition du TST, celui délimité par l'instant de début du premier déplacement et l'instant de fin de la dernière activité. Nous proposons aussi l'opération *Times* :  $TST \rightarrow Alist$  qui retourne la liste des instants temporels *d'intérêts* de la trajectoire. En fait, ces instants sont les instants de début et de fin de chaque activité et de chaque déplacement du TST. La spécification des opérations de projection temporelle est donnée dans le tableau III.22.

**Opérations de recherche temporelle** Ces opérations permettent de retourner une activité, un déplacement ou une liste d'activités ou de déplacements qui se sont déroulés à un instant

Opération	Signature	Description
Temporal_Length	$TST \rightarrow Real$ (III.150)	Retourne l'intervalle de définition du TST
Times	$TST \rightarrow Alist$ (III.151)	Retourne les instants de temps du TST
First_Time	$TST \rightarrow Time$ (III.152)	Retourne le temps de début du TST
Last_Time	$TST \rightarrow Time$ (III.153)	Retourne le temps de fin du TST

**Tableau III.22** – Signature des opérations de projection temporelle

donné ou pendant un intervalle donné. Comme pour les opérations de restriction, quand il s'agit d'un intervalle, il faut que les événements recherchés (activité ou trajectoire) soient temporellement complètement inclus dans l'intervalle en argument. La spécification des opérations de recherche temporelle est donnée dans le tableau III.23.

Opération	Signature	Description
Activity_At_Time	$TST \times Time \rightarrow Activity$ (III.154)	Retourne l'activité qui s'est déroulée à l'instant donné
Trip_At_Time	$TST \times Time \rightarrow Trip$ (III.155)	Retourne le déplacement qui s'est déroulé à l'instant donné
Activities_At_Interval	$TST \times Interval \rightarrow Alist$ (III.156)	Retourne la liste d'activités qui se sont déroulées à l'intervalle donné
Trips_At_Interval	$TST \times Interval \rightarrow Alist$ (III.157)	Retourne la liste des déplacements qui se sont déroulés à l'intervalle donné

**Tableau III.23** – Signature des opérations de recherche temporelle

**Opérations de similarité temporelle** Nous proposons une unique opération de similarité temporelle appelée *Temporal\_Similarity*, de signature (III.158). Cette opération prend en arguments deux TST et retourne un booléen qui indique si ces deux TST sont temporellement similaires. C'est le cas si tous instants d'intérêts des deux TST sont égaux. La spécification de cette opération est donnée dans le tableau III.24.

Opération	Signature	Description
Temporal_Similarity	$TST \times TST \rightarrow Boolean$ (III.158)	Teste si deux TST sont temporellement similaires

Tableau III.24 – Signature de l’opération de similarité temporelle

#### 4.2.6 Opérations spatio-temporelles

Ces opérations ont une sémantique plus générale et associent des opérateurs spatiaux et temporels. A notre connaissance, aucun des travaux cités dans notre état de l’art n’a encore abordé ce genre de manipulation sur les trajectoires spatio-temporelles sémantiquement riches.

**Opérations ensemblistes** Ces opérations sont spécifiées par analogie aux opérations sur les ensembles comme l’égalité, l’union, l’intersection et la différence. Par exemple, l’opération *Equals* de test d’égalité entre trajectoires peut être interprétée comme suit : pour qu’une trajectoire  $tst_1$  soit égale à une trajectoire  $tst_2$ , il faut que les deux trajectoires traduisent le même comportement spatio-temporel et sémantique durant une même durée de temps. Autrement dit, il faut que la succession des déplacements/activités soit similaire en termes de temps de début, temps de fin, en terme de localisations et en termes de nature des activités pour les deux trajectoires. La spécification des opérations de relations temporelles est donnée dans le tableau III.25.

**Opérations de voisinage spatio-temporel** Au sein de la catégorie des opérations spatio-temporelles, nous introduisons des opérations de test de voisinage spatio-temporel dont la spécification est donnée dans le tableau III.25. Ces opérations se présentent sous la forme de prédicats qui permettent de tester la proximité, selon un certain seuil, entre deux TST ou entre un TST et une entité spatiale à un certain instant ou intervalle de temps.

Opération	Signature	Description
<b>Opérations ensemblistes</b>		
Equals	$TST \times TST \rightarrow Boolean$ (III.159)	Teste l'égalité spatio-temporelle entre TST
Includes	$TST \times TST \rightarrow Boolean$ (III.160)	Teste l'inclusion spatio-temporelle entre deux TST
Intersects	$TST \times TST \rightarrow Boolean$ (III.161)	Teste l'intersection spatio-temporelle entre deux TST
Union	$TST \times TST \rightarrow TST$ (III.162)	Retourne l'union de deux TST
Intersection	$TST \times TST \rightarrow TST$ (III.163)	Retourne les TSTs communes entre deux TST
Activities_Intersection	$TST \times TST \rightarrow Alist$ (III.164)	Retourne la liste des activités communes entre deux TST
Trips_Intersection	$TST \times TST \rightarrow Alist$ (III.165)	Retourne la liste des déplacements communs entre deux TST
Difference	$TST \times TST \rightarrow TST$ (III.166)	Retourne la différence (au sens ensembliste) entre deux TST
Activities_Difference	$TST \times TST \rightarrow Alist$ (III.167)	Retourne les activités appartenant à la première TST et non à la deuxième
Trips_Difference	$TST \times TST \rightarrow Alist$ (III.168)	Retourne les déplacements appartenant à la première TST et non à la deuxième
<b>Opérations de voisinage spatio-temporel</b>		
TST_STNeighbour_TST	$TST \times TST \times Real \times Time \rightarrow Boolean$ (III.169)	Teste le voisinage spatio-temporel entre deux TST à un instant donné



TST_STNeighbour_Point	$TST \times Point \times Real \times Time \rightarrow Boolean$ (III.170)	Teste le voisinage spatio-temporel entre un TST et un point à un instant donné
TST_STNeighbour_Polyline	$TST \times Polyline \times Real \times Time \rightarrow Boolean$ (III.171)	Teste le voisinage spatio-temporel entre un TST et une polyligne à un instant donné
TST_STNeighbour_Region	$TST \times Region \times Real \times Time \rightarrow Boolean$ (III.172)	Teste le voisinage spatio-temporel entre un TST et une région à un instant donné

Tableau III.25 – Signature des opérations spatio-temporelles

## 5 Conclusion

Afin d'offrir une modélisation rigoureuse des données spatio-temporelles, il est nécessaire de fixer au préalable les caractéristiques des dimensions spatiales et temporelles utilisées. Ainsi, nous avons défini, dans ce chapitre, le contexte spatial, temporel et sémantique de notre modélisation des trajectoire spatio-temporelle. Pour la dimension sémantique, nous nous sommes appuyés sur le cadre de Time Geography qui perçoit la trajectoire d'objets en mouvement comme un chaînage d'activités et de déplacements. Dans ce cadre, des définitions des notions d'activité, déplacement et de trajectoire spatio-temporelles ainsi que des contraintes sur ces notions ont été proposées. Comme seconde étape et afin de pouvoir proposer un type de données pour la trajectoire spatio-temporelle doté de capacités opérationnelles et fonctionnelles riches et puissantes, nous avons identifié différents besoins d'analyse de cette donnée trajectoire et ceci à différentes dimensions. Autrement dit, pour des dimensions spatiales, temporelles spatio-temporelles et sémantiques, nous avons proposés différentes sortes de manipulation de la trajectoire telle que nous l'avons préalablement définie. Nous avons aussi identifier un besoin relatif à la vérification et au maintien de la cohérence des bases de données manipulant des données du type TST. Cette analyse, nous a permis par la suite, de proposer la spécification d'un large ensemble d'opérations intéressantes pouvant répondre à ses besoins. Cette spécification passe par la présentation des différents types utilisés lors de la définition du nouveau type TST et la définition des signatures des opérations de manipulation du type proposées. Dans la mesure où le type proposé TST est destiné à être intégré dans un SGBD spatial, nous avons supposé l'existence des types abstraits spatiaux habituellement utilisés (tels que *Point*, *Polyline*, *Polygon*) et des types abstraits temporels *Time* et *Interval*. Les opérations topologiques, comme *Meet* ou *Equal* sur ces types seront utilisées pour exprimer l'équivalence avec nos opérateurs.

---

---

# Chapitre IV

---

## Algèbre de définition et de manipulation du TST

### 1 Introduction

La signature précédemment décrite est un modèle syntaxique du type TST. Afin d'exprimer le comportement de cette syntaxe, nous proposons un modèle sémantique sous la forme d'une algèbre associée à cette signature. L'algèbre de définition et de manipulation du type TST est composée de deux parties :

1. *Le domaine du type TST* : Il s'agit de définir le domaine des valeurs du type TST.
2. *La sémantique des opérations* : Il s'agit d'associer à chacune des opérations du type TST une fonction d'interprétation décrite formellement.

Grâce à cette méthode, nous proposons une structure formelle et des mécanismes pour modéliser les dynamiques spatio-temporelles et les sémantiques qui leur sont liées. Nous présentons dans ce qui suit les détails des deux parties composant le modèle sémantique.

### 2 Domaine

Rappelons que dans notre modèle, une activité  $a$  est une entité spatio-temporelle définie par le quadruple :

$$a = (l, t_s, t_e, type) \tag{IV.1}$$

tels  $l \in Point$ ,  $t_s$  et  $t_e \in Time$  et  $type \in String$ . Une condition temporelle essentielle doit être vérifiée pour toute activité. En effet, son temps de début doit être strictement inférieur à son temps de fin. De ce fait, le domaine de valeurs des activités, c'est-à-dire l'ensemble de

toutes les activités pouvant se trouver dans la base de données, est l'ensemble :

$$D_a = \{a \mid a.t_s < a.t_e\} \quad (\text{IV.2})$$

De même un déplacement est formellement défini comme suit :

$$d = (l_s, l_e, t_s, t_e, mode, Path) \quad (\text{IV.3})$$

tels que  $l_s$  et  $l_e \in Point$ ,  $t_s$  et  $t_e \in Time$ ,  $mode \in Alist$  et  $Path \in Polyline$ . Comme pour les activités, un déplacement se déroule pendant une durée de temps non nulle. Le domaine de valeur des déplacements est donc l'ensemble :

$$D_d = \{d \mid d.t_s < d.t_e\} \quad (\text{IV.4})$$

Pour exprimer la succession temporelle des activités d'une trajectoire, nous définissons l'ensemble des  $n$  ( $n \in \mathbb{N}$ ) activités d'un TST comme un sous-ensemble temporellement ordonné de  $D_a$ , nommé  $A$ , tel que :

$$A = \{a_i \mid \forall 1 \leq i < n : a_i \in D_a \wedge (a_i.t_e < a_{i+1}.t_s)\} \quad (\text{IV.5})$$

De même, l'ensemble des  $n$  ( $n \in \mathbb{N}$ ) déplacements d'un TST est l'ensemble temporellement ordonné  $D$ , tel que :

$$D = \{d_i \mid \forall 1 \leq i < n : (d_i \in D_d) \wedge (d_i.t_e < d_{i+1}.t_s)\} \quad (\text{IV.6})$$

Dès lors, le domaine de type TST est l'ensemble des paires  $(A, D)$  tels que :

$$D_{TST} = \{(A, D) \mid \begin{array}{l} (C_1) \quad \forall 1 < i \leq n : (d_i.l_s = a_{i-1}.l) \wedge (d_i.t_s = a_{i-1}.t_e + 1) \\ (C_2) \quad \forall 1 \leq i \leq n : (d_i.l_e = a_i.l) \wedge (d_i.t_e = a_i.t_s - 1) \end{array}\} \quad (\text{IV.7})$$

Dans cette formulation du domaine, les deux premières conditions annotées par (C1) correspondent aux contraintes spatio-temporelles (1) et (3) décrites dans l'ensemble d'équations (III.1) du chapitre précédent. Les conditions annotées par (C2) correspondent aux contraintes (2) et (4). Ces contraintes, accompagnées des relations d'ordre temporel sur les activités et les déplacements, traduisent le cheminement des événements au sein du TST. La relation d'ordre permet également d'établir un lien sémantique entre un déplacement d'ordre  $i$  et l'activité de même ordre qui représente l'objectif de ce déplacement.

### 3 Sémantique des opérations

Nous allons, par la suite, donner les fonctions d'interprétation des opérations sur le TST, décrits par leur signature dans le chapitre précédent. Ces fonctions seront exprimées sur le domaine du type sous la forme :

$$f_{NomOpération}(param_1, \dots, param_n) := interprétation$$

Pour une meilleur lecture, nous associons dans notre description de ces fonctions le numéro relatif à la signature de l'opération associée. La formulations mathématique de la sémantique des opérations de manipulation du TST représente une description du comportement de ces opérations. Dans ce mémoire, nous associons, en Annexe A, aux opérations dont la sémantique est complexe des algorithmes qui décrivent plus en détail cette sémantique. Ces algorithmes permettront une intégration aisée et pertinente du nouveau type proposé TST au sein de prototypes informatiques manipulant les trajectoires spatio-temporelles et ayant recours à la manipulation de l'aspect sémantique du mouvement.

#### 3.1 Opérations de vérification de la cohérence de la base de données

La sémantique de ces opérations est illustrée dans le tableau IV.1.

Signature	Fonction d'interprétation
(III.2)	$f_{check\_activity}(a) := a.t_s < a.t_e \wedge a.l \in Point$ (IV.8)
(III.3)	$f_{check\_trip}(d) := d.t_s < d.t_e \wedge d.l_s \in Point \wedge d.l_e \in Point \wedge d.path \in Polyline$ (IV.9)
(III.4)	$f_{existe\_tst}(t, tst) := tst.d_1.t_s \leq t \wedge t \leq tst.a_n.t_e$ (IV.10)
(III.5)	$f_{check\_tst}(tst) := (\forall 1 \leq i \leq n : check\_activity(a_i) \wedge check\_trip(d_i) \wedge d_i.l_e = a_i.l \wedge d_i.t_e = a_i.t_s - 1) \wedge (\forall 1 < i \leq n : d_i.l_s = a_{i-1}.l \wedge d_i.t_s = a_{i-1}.t_e + 1)$ (IV.11)

**Tableau IV.1** – Sémantique des opérations de vérification de la cohérence de la base de données

En effet, l'opération *check\_activity* vérifie qu'un objet de type *Activity* est valide temporellement et spatialement. D'un point de vue temporel, il suffit de vérifier que le temps de début d'une activité,  $a \in Activity$ , est strictement inférieur à son temps de fin. Spatialement, il s'agit de vérifier si la valeur de sa localisation spatiale appartient bien au type *Point*. La sémantique de cette opération est exprimée par la fonction (IV.8).

L'opération *check\_trip* vérifie qu'un objet de type *Trip* est valide temporellement et spatialement. D'un point de vue temporel, il suffit de vérifier que le temps de début d'un déplacement,

$d \in Trip$ , est strictement inférieur à son temps de fin. Spatialement, il s'agit de vérifier si les valeurs de ses localisations spatiales de départ et d'arrivée appartiennent bien au type *Point* et que la valeur de son attribut *Path* appartient bien au type *Polyline*. Ceci est formulé par la fonction (IV.9).

L'opération *existe\_tst* vérifie qu'à un instant de temps  $t \in Instant$ , une valeur *tst* de type TST est bien définie. La fonction (IV.10) exprime cette sémantique.

L'opération *check\_tst* vérifie la validité d'une valeur *tst* de type TST. Sa sémantique est exprimée par la fonction (IV.11). En effet, une valeur de type TST est valide si les activités et les déplacements la composant sont valides et si les contraintes spatio-temporelles assurant le chaînage des événements dans le TST sont vérifiées.

### 3.2 Les créateurs

La sémantique de ces opérations est illustrée dans le tableau IV.2. En effet, l'interprétation de l'opération *Make*, qui crée une trajectoire vide, s'exprime simplement dans la la fonction (IV.12), par la création d'un couple d'ensembles vides, l'un relatif aux activités et l'autre relatif aux déplacements.

L'ajout d'un couple de déplacement/activité,  $(d,a)$ , dans la trajectoire *tst* s'exprime sémantiquement par des opérations d'union entre ensembles. Cet ajout doit obeir à des conditions qui garantissent le chaînage spatio-temporel dans la trajectoire. Cette sémantique est illustrée par la la fonction (IV.13).

La suppression dans la trajectoire s'exprime, comme le montre la fonction (IV.14), par deux opérations de différence d'une part entre l'ensemble  $A$  des activités et l'ensemble constitué par la dernière activité de  $A$  et d'autre part entre l'ensemble  $D$  des déplacements et l'ensemble constitué par le dernier déplacement de  $D$ . Ceci est vrai à condition que les ensembles  $A$  et  $B$  soient non vides.

Signature	Fonction d'interprétation
(III.6)	$f_{Make}() := (\emptyset, \emptyset)$ (IV.12)
(III.7)	$f_{Add}(tst, d, a) := (A \cup \{a\}, D \cup \{d\})$ if $(d.l_s = a_n.l) \wedge (d.l_e = a.l) \wedge (d.t_s = a_n.t_e + 1) \wedge (d.t_e = a.t_s - 1)$ (IV.13)
(III.8)	$f_{Sub}(tst) := (A \setminus \{a_n\}, D \setminus \{d_n\})$ if $(card(A) \neq 0) \wedge (card(D) \neq 0)$ (IV.14)

Tableau IV.2 – Sémantique des créateurs

### 3.3 Opérations sémantiques

Dans ce qui suit, nous décrivons mathématiquement le comportement des opérations sémantiques sur le TST.

#### 3.3.1 Opérations de comptage sémantique

La sémantique de la totalité des opérations de comptage sémantique est illustrée dans le tableau IV.3. En effet, l'interprétation de ces opérations qui effectuent un comptage sur les événements du TST est relativement simple. En effet, étant donné une valeur  $tst \in D_{TST}$ , le calcul du nombre d'éléments de cette trajectoire se limite à déterminer la cardinalité soit de l'ensemble,  $A$ , de ses activités, soit de l'ensemble,  $D$ , de ses déplacements, vu qu'il y a autant de déplacements que d'activités dans un TST. Les fonctions (IV.15) et (IV.16) illustrent mathématiquement cette sémantique.

Soit les valeurs  $tst \in D_{TST}$  et  $m \in String$ . Les calculs du nombre d'activités et du nombre de déplacements exercés selon un mode de déplacement indiqué sont décrits similairement. C'est la cardinalité d'un sous-ensemble de déplacements appartenant à la valeur  $tst$ . Ces déplacements doivent avoir été exercés selon le mode de déplacement  $m$  indiqué. L'interprétation mathématique de ces opérations est donnée par les fonctions (IV.17) et (IV.18).

De même, les calculs du nombre d'activités et du nombre de déplacements, exercés selon un objectif indiqué, sont décrits similairement. C'est la cardinalité d'un ensemble d'activités appartenant à  $tst$ . Ces activités doivent avoir comme type le type  $m$  indiqué (voir les fonctions (IV.19) et (IV.20)).

Signature	Fonction d'interprétation
(III.9)	$f_{Trip\_Count}(tst) := card(D)$ (IV.15)
(III.10)	$f_{Activity\_Count}(tst) := card(A)$ (IV.16)
(III.11)	$f_{Activities\_With\_Mode\_Count}(tst, m) := x \in N : x = card(\{d \in D   m \in d.mode\})$ (IV.17)
(III.14)	$f_{Trips\_With\_Mode\_Count}(tst, m) := x \in N : x = card(\{d \in D   m \in d.mode\})$ (IV.18)
(III.12)	$f_{Activities\_With\_Type\_Count}(tst, m) := x \in N : x = card(\{a \in A   m = a.type\})$ (IV.19)
(III.13)	$f_{Trips\_With\_Purpose\_Count}(tst, m) := x \in N : x = card(\{a \in A   m = a.type\})$ (IV.20)

Tableau IV.3 – Sémantique des opérations de comptage sémantique

## 3.3.2 Opérations de positionnement

Nous rappelons que le premier lot de ce genre d'opérations retourne un unique événement. Cet événement peut être sélectionné selon sa position absolue dans le chaînage du TST ou selon sa position relative au sein de ce même chaînage et ceci par rapport à un autre événement. L'interprétation de la sémantique de la totalité de ces opérations est exprimée dans le tableau IV.4.

Signature	Fonction d'interprétation
(III.15)	$f_{First\_Trip}(tst) := d_1$ (IV.21)
(III.16)	$f_{First\_Activity}(tst) := a_1$ (IV.22)
(III.17)	$f_{Last\_Trip}(tst) := d_n$ (IV.23)
(III.18)	$f_{Last\_Activity}(tst) := a_n$ (IV.24)
(III.20)	$f_{Nth\_Activity}(tst, i) := a_i \in A$ if $1 \leq i \leq n$ (IV.25)
(III.19)	$f_{Nth\_Trip}(tst, i) := d_i \in D$ if $1 \leq i \leq n$ (IV.26)
(III.21)	$f_{Activity\_Before\_Trip}(tst, d) := a_i \in A$ if $\exists 1 \leq i < n : d = d_{i+1}$ (IV.27)
(III.22)	$f_{Activity\_After\_Trip}(tst, d) := a_i \in A$ if $\exists 1 \leq i \leq n : d = d_i$ (IV.28)
(III.23)	$f_{Trip\_Before\_Activity}(tst, a) := d_i \in D$ if $\exists 1 \leq i \leq n : a = a_i$ (IV.29)
(III.24)	$f_{Trip\_After\_Activity}(tst, a) := d_i \in D$ if $\exists 1 < i \leq n : a = a_{i-1}$ (IV.30)
(III.25)	$f_{Trip\_Before\_Trip}(tst, d) := d_i \in D$ if $\exists 1 \leq i < n : d = d_{i+1}$ (IV.31)
(III.26)	$f_{Trip\_After\_Trip}(tst, d) := d_i \in D$ if $\exists 1 < i \leq n : d = d_{i-1}$ (IV.32)
(III.27)	$f_{Activity\_Before\_Activity}(tst, a) := a_i \in A$ if $\exists 1 \leq i < n : a = a_{i+1}$ (IV.33)
(III.28)	$f_{Activity\_After\_Activity}(tst, a) := a_i \in A$ if $\exists 1 < i \leq n : a = a_{i-1}$ (IV.34)

**Tableau IV.4** – Sémantique des opérations de positionnement sémantique retournant un événement

Dans le premier cas, les fonctions *First\_Activity* et *Last\_Activity* (respectivement *First\_Trip* et *Last\_Trip*) retournent les activités de l'ensemble ordonné  $A$  (respectivement les déplacements de l'ensemble ordonné  $D$ ) se trouvant aux positions 1 et  $n$ . Ceci est traduit mathématiquement par les fonctions (IV.22), (IV.24), (IV.21) et (IV.23). Étant donné une valeur de trajectoire  $tst$  et un entier  $i$ , la fonction *Nth\_Activity*( $tst, i$ ) (respectivement *Nth\_Trip*( $tst, i$ )), quant à elle, retourne l'activité (respectivement le déplacement) de la trajectoire  $tst$  se trouvant à la position  $i$  à condition que l'entier  $i$  soit compris entre 1 et  $n$ . Ceci est exprimé par les fonctions (IV.25) et (IV.26).

Dans le cas où la sélection de l'événement se fait selon un critère de positionnement relatif, et



## Chapitre IV. Algèbre de définition et de manipulation du TST

---

pour exprimer la relation de précédence/succession entre événement, nous exploitons la relation d'ordre qui existe dans les ensembles  $A$  et  $D$  du TST. À titre d'exemple, étant donné une trajectoire  $tst$  et un déplacement  $d$ , la fonction  $Activity\_After\_Trip(tst, d)$  retourne l'activité qui suit immédiatement le déplacement  $d$ . Vu que l'ajout dans la trajectoire se fait par couple (déplacement/activité), un déplacement et l'activité qui le suit ont la même position dans la trajectoire. La fonction doit donc retourner l'activité qui figure, dans l'ensemble ordonné  $A$ , à la même position à laquelle figure  $d$  dans l'ensemble ordonné  $D$ . Ceci est illustré dans la fonction (IV.28). Les autres fonctions de positionnement relatif sont exprimées sur la base de ce même raisonnement.

Nous avons aussi proposé un lot d'opérations de positionnement qui retournent un ensemble d'événements qui suivent ou précèdent un événement donné. À titre d'exemple, étant donné une valeur de trajectoire  $tst \in D_{tst}$  et un déplacement  $d \in D$ , l'opération  $Activities\_After\_Trip(tst, d)$  retourne un sous ensemble ordonné de l'ensemble  $A$ . Ce sous ensemble a comme premier élément l'activité qui suit immédiatement  $d$  dans la trajectoire et qui est donc à la même position que lui (voir la fonction (IV.36)). Le reste des opérations de ce lot sont définis en suivant le même raisonnement. La formulation mathématique de la sémantique de toutes ces opérations est exprimée dans le tableau IV.5.

Signature	Fonction d'interprétation
(III.29)	$f_{Activities\_Before\_Trip}(tst, d) := \{a_i \in A   1 \leq i \leq max\} \text{ if } (d \neq d_1)$ $\wedge (\exists 1 \leq max < n : d = d_{max+1})$ <span style="float: right;">(IV.35)</span>
(III.30)	$f_{Activities\_After\_Trip}(tst, d) := \{a_i \in A   min \leq i \leq n\} \text{ if } (d \neq d_n)$ $\wedge (\exists 1 \leq min \leq n : d = d_{min})$ <span style="float: right;">(IV.36)</span>
(III.31)	$f_{Trips\_Before\_Activity}(tst, a) := \{d_i \in D   1 \leq i \leq max\} \text{ if } (a \neq a_n)$ $\wedge (\exists 1 \leq max \leq n : a = a_{max})$ <span style="float: right;">(IV.37)</span>
(III.32)	$f_{Trips\_After\_Activity}(tst, a) := \{d_i \in D   min \leq i \leq n\} \text{ if } (a \neq a_n)$ $\wedge (\exists 1 < min \leq n : a = a_{min-1})$ <span style="float: right;">(IV.38)</span>
(III.33)	$f_{Trips\_Before\_Trip}(tst, d) := \{d_i \in D   1 \leq i \leq max\} \text{ if } (d \neq d_1)$ $\wedge (\exists 1 \leq max < n : d = d_{max+1})$ <span style="float: right;">(IV.39)</span>
(III.34)	$f_{Trips\_After\_Trip}(tst, d) := \{d_i \in D   min \leq i \leq n\} \text{ if } (d \neq d_n)$ $\wedge (\exists 1 < min \leq n : d = d_{min-1})$ <span style="float: right;">(IV.40)</span>

(III.35)	$f_{Activities\_Before\_Activity}(tst, a) := \{a_i \in A   1 \leq i \leq max\} \text{ if } (a \neq a_1)$ $\wedge (\exists 1 \leq max < n : a = a_{max+1})$ <span style="float: right;">(IV.41)</span>
(III.36)	$f_{Activities\_After\_Activity}(tst, a) := \{a_i \in A   min \leq i \leq n\} \text{ if } (a \neq a_n)$ $\wedge (\exists 1 < min \leq n : a = a_{min-1})$ <span style="float: right;">(IV.42)</span>

**Tableau IV.5** – Sémantique des opérations de positionnement sémantique retournant une liste d'événements

Enfin, étant donné une valeur  $tst \in D_{tst}$ , une activité  $a \in A$  et un déplacement  $d \in D$ , la fonction  $Position\_Activity(tst, a)$  (respectivement  $Position\_Trip(tst, d)$ ) retourne un entier non nul  $i$ , si cet entier existe comme position dans la trajectoire et si l'activité (respectivement le déplacement) à cette position est identique à l'activité  $a$  (respectivement  $d$ ). Ceci est exprimé mathématiquement par les fonctions (IV.43) et (IV.44).

Signature	Fonction d'interprétation
(III.37)	$f_{Position\_Activity}(tst, a) := i \in N^* \text{ if } \exists a_i \in A : a = a_i$ (IV.43)
(III.38)	$f_{Position\_Trip}(tst, d) := i \in N^* \text{ if } \exists d_i \in D : d = d_i$ (IV.44)

**Tableau IV.6** – Sémantique des opérations de positionnement sémantique retournant la position d'un événement dans le TST

### 3.3.3 Opérations de recherche sémantique

Nous proposons ici la sémantique du groupe d'opérations de recherche sémantique qui retournent un ensemble d'événements respectant un critère donné. À titre d'exemple, étant donné une valeur de trajectoire  $tst$  et une valeur  $m \in String$ , l'appel à l'opération  $Trips\_With\_Purpose(tst, m)$  retourne l'ensemble des déplacements de la valeur  $tst$  qui ont comme but la chaîne de caractère  $m$ . Dans notre conception des choses, le but d'un déplacement est la nature de l'activité qui le suit. Donc, les déplacements retournés doivent avoir chacun comme activité suivante une activité de l'ensemble  $A$  qui a comme nature la valeur  $m$ . La sémantique du reste de ce groupe de fonctions s'exprime mathématiquement dans le tableau IV.7

Nous avons aussi proposé comme opérations de recherche sémantique un second groupe qui permet la recherche et la navigation dans la trajectoire selon le critère sémantique. Ce groupe comprend des opérations qui, étant donné une trajectoire et un critère, retournent le premier

Signature	Fonction d'interprétation
(III.56)	$f_{Activities\_With\_Type}(tst, m) := \{a \in A   m = a.type\}$ (IV.45)
(III.57)	$f_{Trips\_With\_Mode}(tst, m) := \{d \in D   m \in d.mode\}$ (IV.46)
(III.58)	$f_{Trips\_With\_Purpose}(tst, m) := \{d \in D   \exists a \in A : a.type = m$ $\wedge d = Trip\_Before\_Activity(tst, a)\}$ (IV.47)
(III.55)	$f_{Activities\_With\_Mode}(tst, m) := \{a \in A   \exists d \in D : m \in d.mode$ $\wedge a = Activity\_After\_Trip(tst, d)\}$ (IV.48)

**Tableau IV.7** – Sémantique des opérations de recherche sémantique retournant une liste d'événements

ou dernier événement qui vérifie ce critère. Étant donné maintenant une trajectoire, un critère et un événement, nous proposons aussi des fonctions qui retournent l'événement qui suit ou précède l'événement en argument (pas obligatoirement directement) et qui vérifie le critère spécifié. À titre d'exemple, pour l'interprétation de l'opération *First\_Activity\_With\_Type* (voir equation (IV.49)), nous nous sommes servi de fonctions que nous avons déjà proposé comme les fonction *Activity\_with\_Type\_Count*, *At\_Activity\_Activity* et *First\_Activity*. Ainsi, étant donné une valeur de trajectoire  $tst \in D_{TST}$  et un critère  $m \in String$  l'appel *First\_Activity\_With\_Type*( $tst, m$ ) retourne une activité  $a$ , dont la nature est égale à la valeur de  $m$ . De plus,  $a$  doit être telle que la "sous-trajectoire", résultat de la restriction de la trajectoire  $tst$  de la première activité à l'activité  $a$ , ne possède dans son chaînage qu'une seule activité qui a comme nature le type en argument  $m$ . Cette activité est forcément  $a$  puisque la valeur de l'attribut type de  $a$  est égal à  $m$ .

Dans la figure IV.1, si  $m = "shopping"$ , la première activité ayant ce type est l'activité  $a_2$ . La trajectoire restreinte de la première activité de la trajectoire à cette activité est indiquée en bleu. L'activité  $a_2$  est la seule ayant ce type dans la trajectoire restreinte (en bleu) car le nombre d'activités de type "shopping" est égal à 1. L'appel *First\_Activity\_With\_Type*( $tst, "shpping"$ ) retournera donc l'activité  $a_2$ .

La sémantique du reste de cette catégorie d'opérations est exprimée dans le tableau IV.8

### 3.3.4 Opérations de restriction sémantique

L'interprétation de ce type d'opérations est récapitulé dans le tableau IV.9. Soit  $a$  et  $aa$  deux valeurs d'activité  $\in D_A$  et  $tst$  une valeur de TST  $\in D_{TST}$ . La restriction sémantique, *At\_Activity\_Activity*, de la trajectoire  $tst$ , délimitée par les deux activités  $a$  et  $aa$ , est un couple formé par deux sous ensembles temporellement ordonnés d'activités et de déplacements.

Signature	Fonction d'interprétation
(III.39)	$f_{First\_Activity\_With\_Type}(tst, m) := a \in A : a.type = m \wedge Activities\_With\_Type\_Count(At\_Activity\_Activity(tst, First\_Activity(tst, a), m) = 1$
(III.40)	$f_{Last\_Activity\_With\_Type}(tst, m) := a \in A : a.type = m \wedge Activities\_With\_Type\_Count(At\_Activity\_Activity(tst, a, Last\_Activity(tst)), m) = 1$
(III.41)	$f_{Next\_Activity\_With\_Type}(tst, a, m) := e \in A : e = First\_Activity\_With\_Type(At\_Activity\_Activity(tst, Next\_Activity(tst, a), Last\_Activity(tst)), m)$
(III.42)	$f_{Previous\_Activity\_With\_Type}(tst, a, m) := e \in A : e = Last\_Activity\_With\_Type(At\_Activity\_Activity(tst, First\_Activity(tst), Previous\_Activity(tst, a)), m)$
(III.43)	$f_{First\_Trip\_With\_Purpose}(tst, m) := d \in D \text{ if } \exists a \in A : a.type = m \wedge Activities\_With\_Type\_Count(At\_Activity\_Activity(tst, First\_Activity(tst), a), m) = 1 \wedge d = Trip\_Before\_Activity(tst, a)$
(III.44)	$f_{Last\_Trip\_With\_Purpose}(tst, m) := d \in D \text{ if } \exists a \in A : a.type = m \wedge Activities\_With\_Type\_Count(At\_Activity\_Activity(tst, a, Last\_Activity(tst)), m) = 1 \wedge d = Trip\_Before\_Activity(tst, a)$
(III.45)	$f_{Next\_Trip\_With\_Purpose}(tst, d, m) := e \in D \text{ if } \exists a \in A : a = First\_Activity\_With\_Type(At\_Trip\_Trip(tst, Next\_Trip(tst, d), Last\_Trip(tst)), m) \wedge a = Activity\_After\_Trip(tst, e)$
(III.46)	$f_{Previous\_Trip\_With\_Purpose}(tst, d, m) := e \in D \text{ if } \exists a \in A : a = Last\_Activity\_With\_Type(At\_Trip\_Trip(tst, First\_Trip(tst), Previous\_Trip(tst, d)), m) \wedge a = Activity\_After\_Trip(tst, e)$
(III.47)	$f_{First\_Trip\_With\_Mode}(tst, m) := d \in D : m \in d.type \wedge Trips\_With\_Mode\_Count(At\_Trip\_Trip(tst, First\_Trip(tst), d), m) = 1$
(III.48)	$f_{Last\_Trip\_With\_Mode}(tst, m) := d \in D :  m \in d.mode \wedge Trips\_With\_Mode\_Count(At\_Trip\_Trip(tst, d, Last\_Trip(tst)), m) = 1$

(III.49)	$f_{Next\_Trip\_With\_Mode}(tst, d, m) := e \in D : e = First\_Trip\_With\_Mode(At\_Trip\_Trip(tst, Next\_Trip(tst, d), Last\_Trip(tst)), m) \quad (IV.59)$
(III.50)	$f_{Previous\_Trip\_With\_Mode}(tst, d, m) := e \in D : e = Last\_Trip\_With\_Mode(At\_Trip\_Trip(tst, First\_Trip(tst), Previous\_Trip(tst, d)), m) \quad (IV.60)$
(III.51)	$f_{First\_Activity\_With\_Mode}(tst, m) := a \in Ai.f \exists d \in D : m \in d.mode \wedge Trips\_With\_Mode\_Count(At\_Trip\_Trip(tst, First\_Trip(tst), d) = 1 \wedge d = Trip\_Before\_Activity(tst, a) \quad (IV.61)$
(III.52)	$f_{Last\_Activity\_With\_Mode} := a \in Ai.f \exists d \in D : m \in d.mode \wedge Trips\_With\_Mode\_Count(At\_Trip\_Trip(tst, d, Last\_Trip(tst)) = 1 \wedge d = Trip\_Before\_Activity(tst, a) \quad (IV.62)$
(III.54)	$f_{Previous\_Activity\_With\_Mode}(tst, a, m) := e \in Ai.f \exists d \in D : d = Trip\_Before\_Activity(tst, e) \wedge d = Last\_Trip\_With\_Mode(At\_Activity\_Activity(tst, First\_Activity(tst), Previous\_Activity(tst, a), m) \quad (IV.63)$
(III.53)	$f_{Next\_Activity\_With\_Mode}(tst, a, m) := e \in Ai.f \exists d \in D : d = First\_Trip\_With\_Mode(At\_Activity\_Activity(tst, Next\_Activity(tst, a), Last\_Activity(tst)), m) \wedge d = Trip\_Before\_Activity(tst, e) \quad (IV.64)$

**Tableau IV.8** – Sémantique des opérations de recherche sémantique retournant un événement



Signature	Fonction d'interprétation
(III.61)	$f_{Activity\_List}(tst) := A$ (IV.67)
(III.62)	$f_{Trip\_List}(tst) := D$ (IV.68)

Tableau IV.10 – Sémantique des opérations de projection dans le domaine sémantique

### 3.3.6 Opérations de similarité sémantique

Étant donné deux valeur  $tst_1$  et  $tst_2 \in D_{TST}$ , ces deux trajectoires sont dites sémantiquement similaires si leur ensembles d'activités sont de même cardinalité. De plus, il faut qu'au sein des ensembles ordonnés des activités et des déplacements de ces trajectoires, toute activité de  $tst_1$  soit de même nature que l'activité qui lui correspond (même ordre dans le chainage) dans  $tst_2$ . Il est aussi nécessaire de vérifier que tout déplacement de  $tst_1$  possède les mêmes modes de déplacement que le déplacement qui lui correspond dans  $tst_2$ . La sémantique de l'opération *Semantic\_Similarity* s'exprime, alors, par la fonction (IV.69).

Signature	Fonction d'interprétation
(III.63)	$f_{Semantic\_Similarity}(tst_1, tst_2) := (card(tst_1.A) = card(tst_2.A)) \wedge (\forall 1 \leq i \leq n, \\ tst_1.A.a_i.type = tst_2.A.a_i.type \wedge tst_1.D.d_i.mode = tst_2.D.d_i.mode)$ (IV.69)

Tableau IV.11 – Sémantique de l'opération de test de la similarité sémantique

### 3.3.7 Prédicats sémantiques

Les prédicats d'appartenance à un TST s'expriment par des tests d'appartenance simple aux ensembles  $A$  et  $D$ . Le tableau IV.12 illustre l'interprétation mathématique de ces opérations.

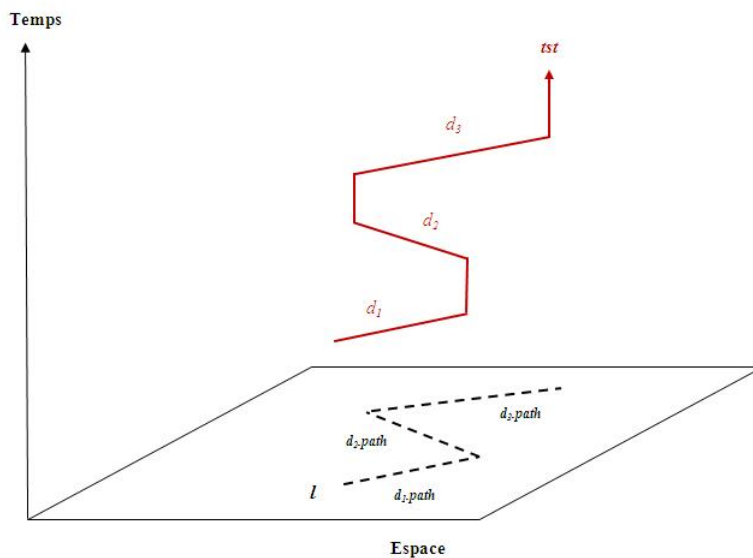
Signature	Fonction d'interprétation
(III.65)	$f_{Include\_Trip}(tst, d) := (d \in D)$ (IV.70)
(III.64)	$f_{Include\_Activity}(tst, a) := (a \in A)$ (IV.71)
(III.66)	$f_{Empty}(tst) := (card(A) = 0) \wedge (card(D) = 0)$ (IV.72)

Tableau IV.12 – Sémantique des opérations de type prédicats sémantiques

### 3.4 Opérations spatiales

#### 3.4.1 Opérations de projection dans le domaine spatial

Lorsqu'il s'agit d'une projection spatiale totale, l'équation (IV.73) indique que l'itinéraire total parcouru par l'objet mobile, au sein d'une trajectoire  $tst \in D_{tst}$  est un polyligne  $l \in Polyline$ . La valeur de  $l$  est obtenue par l'application d'une fonction spatiale classique d'agrégation de lignes reliées spatialement. Cette fonction est appliquée aux chemins  $d_i.path$ , relatifs à tous les déplacements  $d_i$ , de la trajectoire  $tst$  (voir figure IV.2). Cette agrégation est dans notre cas une forme de concaténation spatiale des ces chemins qui est toujours possible, vu le caractère chaîné et les conditions spatiales que vérifie une valeur de type TST (voir conditions (1) et (2) dans l'équation (III.1) du chapitre 3).



**Figure IV.2** – Itinéraire spatial d'un TST

Lorsqu'il s'agit d'une projection spatiale partielle, la sémantique des opérations est exprimée par les équations (IV.74), (IV.75), (IV.76) et (IV.77).

En effet, étant donné une valeur  $tst \in D_{tst}$ ,  $x$  et  $y$  deux activités appartenant à l'ensemble  $A$  de la valeur  $tst$ , si  $x$  précède temporellement  $y$  alors le chemin entre  $x$  et  $y$  ( $Path\_between(tst, x, y)$ ) est l'itinéraire total parcouru dans la sous trajectoire résultat de la restriction de la trajectoire  $tst$  de l'activité qui suit  $x$  vers  $y$ . Ceci est exprimé par la fonction (IV.74).

À titre d'exemple, dans la figure IV.3 et au sein de la trajectoire  $tst$ , composée de quatre activités, le chemin qui mène de l'activité  $a_2$  à l'activité  $a_4$  est l'itinéraire spatial de la sous-trajectoire  $tst'$  mentionnée en bleu. Cette sous-trajectoire n'est autre que la restriction de  $tst$



## Chapitre IV. Algèbre de définition et de manipulation du TST

---

de l'activité  $a_3$  (activité suivante de  $a_2$ ) à l'activité  $a_4$ .

L'ensemble des points spatiaux d'une trajectoire est donné par l'ensemble des valeurs des localisations de ses activités. Ceci est exprimé par la fonction (IV.75).

Le premier point spatial d'une valeur de trajectoire,  $tst \in D_{TST}$ , est le point de départ du premier déplacement, qui est  $d_1$ , au sein de la trajectoire. Le dernier point spatial d'une trajectoire est la localisation de la dernière activité, qui est  $a_n$ , au sein de la trajectoire. Ceci est exprimé par les fonctions (IV.76) et (IV.77).

La sémantique de l'ensemble de ses opérations est récapitulée dans le tableau IV.13.

Signature	Fonction d'interprétation
(III.67)	$f_{Itinerary}(tst) := (d_1.path \text{ if } card(D) = 1) \vee (l \in Polyline, l = concat(< d_1.path, \dots, d_n.path >))$ (IV.73)
(III.68)	$f_{Path\_Between}(tst, x, y) := (Itinerary(At\_Activity\_Aactivity(Activity\_After\_Activity(tst, x), y)) \text{ if } x, y \in A \wedge x.t_e < y.t_s) \vee (Itinerary(At\_Activity\_Activity(Activity\_After\_Activity(tst, y), x)) \text{ if } x, y \in A \wedge y.t_e < x.t_s)$ (IV.74)
(III.69)	$f_{Points}(tst) := \{a.l \in Point, \forall a \in A\}$ (IV.75)
(III.70)	$f_{First\_Point}(tst) := d_1.l_s \in Point$ (IV.76)
(III.71)	$f_{End\_Point}(tst) := a_n.l \in Point$ (IV.77)

**Tableau IV.13** – Sémantique des opérations de projection dans le domaine spatial

### 3.4.2 Opérations de relations spatiales

La sémantique des opérations de topologie spatiale sur le type TST est définie à l'aide des relations spatiales entre objets géométriques définies dans [21, 23]. Des relations d'équivalence spatiales vont permettre d'exprimer ces opérations.

Pour le domaine spatial, une variable  $tst$  de type TST est traitée comme étant une polyligne, résultat de l'opération  $Itinerary(tst)$ . Cette polyligne représente le trajet parcouru par une personne au cours de ces différents déplacements. Ainsi, tester si un TST croise une polyligne  $l$  revient à évaluer le prédicat :  $Itinerary(tst) \text{ Cross } l$ . Le même raisonnement est appliqué pour les autres opérations de topologies spatiales. La sémantique de ces opérations est illustrée dans le tableau IV.14.

### IV.3 Sémantique des opérations

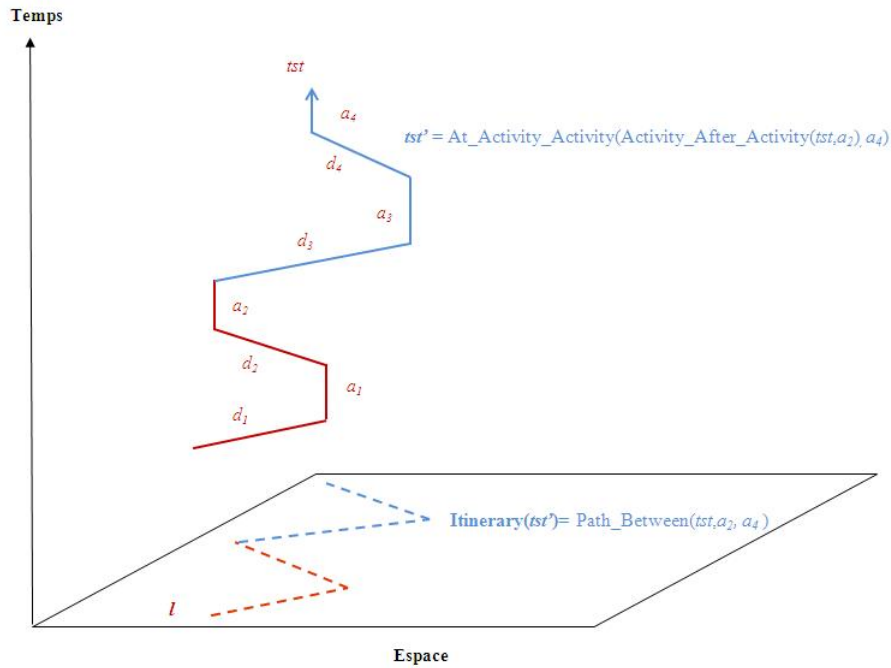


Figure IV.3 – Chemin spatial entre deux activités au sein d'un TST

Signature	Fonction d'interprétation
(III.90)	$f_{TST\_Contains\_Point}(p, tst) := Itinerary(tst) \text{ Contains } p$ (IV.78)
(III.91)	$f_{TST\_EndsBy\_Point}(tst, p) := Itinerary(tst) \text{ EndsBy } Point$ (IV.79)
(III.92)	$f_{TST\_BeginsBy\_Point}(tst, p) := Itinerary(tst) \text{ BeginsBy } p$ (IV.80)
(III.93)	$f_{TST\_Equal\_Polyline}(tst, l) := Itinerary(tst) \text{ Equal } l$ (IV.81)
(III.94)	$f_{TST\_Cross\_Polyline}(tst, l) := Itinerary(tst) \text{ Cross } l$ (IV.82)
(III.95)	$f_{TST\_Disjoint\_Polyline}(tst, l) := Itinerary(tst) \text{ Disjoint } l$ (IV.83)
(III.96)	$f_{TST\_Contains\_Polyline}(tst, l) := Itinerary(tst) \text{ Contains } l$ (IV.84)
(III.97)	$f_{TST\_Inside\_Region}(tst, r) := Itinerary(tst) \text{ Inside } r$ (IV.85)
(III.98)	$f_{TST\_Touch\_Region}(tst, r) := Itinerary(tst) \text{ Touch } r$ (IV.86)
(III.99)	$f_{TST\_Cross\_Region}(tst, r) := Itinerary(tst) \text{ Cross } r$ (IV.87)
(III.100)	$f_{TST\_Disjoint\_Region}(tst, r) := Itinerary(tst) \text{ Disjoint } r$ (IV.88)
(III.101)	$f_{TST\_SpatialEqual\_TST}(tst_1, tst_2) := Itinerary(tst_1) \text{ Equal } Itinerary(tst_2)$ (IV.89)

(III.103)	$f_{TST\_SpatialCross\_TST}(tst_1, tst_2) := Itinerary(tst_1) \text{ Cross } Itinerary(tst_2)$ (IV.90)
(III.104)	$f_{TST\_SpatialContains\_TST}(tst_1, tst_2) := Itinerary(tst_1) \text{ Contains } Itinerary(tst_2)$ (IV.91)

**Tableau IV.14** – Sémantique des opérations de topologie spatiale

### 3.4.3 Opérations de voisinage spatial

Étant donné deux valeurs de trajectoires  $tst_1$  et  $tst_2 \in TST$  et une distance spatiale  $s \in Real$ , tester si ces deux trajectoires sont proches l'une de l'autre selon la distance  $s$ , revient à tester si leur itinéraires spatiales le sont. Il suffit alors d'appliquer une relation spatiale classique de test de proximité entre objets géométriques, que nous nommons ici *Neighbour* entre les deux itinéraires. La sémantique de ces opérations est introduite dans le tableau IV.15.

Signature	Fonction d'interprétation
(III.105)	$f_{TST\_SpatialNeighbour\_TST}(tst_1, tst_2, s) := Neighbour(Itinerary(tst_1), Itinerary(tst_2), s)$ (IV.92)
(III.106)	$f_{TST\_SpatialNeighbour\_Point}(tst, p, s) := Neighbour(Itinerary(tst), p, s)$ (IV.93)
(III.107)	$f_{TST\_SpatialNeighbour\_Polyline}(tst, l, s) := Neighbour(Itinerary(tst), l, s)$ (IV.94)
(III.108)	$f_{TST\_SpatialNeighbour\_Region}(tst, r, s) := Neighbour(Itinerary(tst), r, s)$ (IV.95)

**Tableau IV.15** – Sémantique des opérations de voisinage spatial

### 3.4.4 Opérations de calcul spatial

La sémantique de ce type d'opérations est exprimée par des fonctions d'interprétation que nous avons regroupée dans le tableau IV.16.

Pour calculer la longueur spatiale totale d'une valeur de trajectoire  $tst \in D_{TST}$ , il suffit d'appliquer un opérateur classique de calcul de longueur spatiale d'une entité géométrique (éventuellement un ligne), appelé ici *Length*, à l'itinéraire total de la valeur  $tst$ . La fonction (IV.96) exprime ceci.

Signature	Fonction d'interprétation
(III.109)	$f_{Spatial\_Length}(tst) := Length(Itinerary(tst))$ (IV.96)
(III.110)	$f_{Length\_Path\_Between}(tst, x, y) := (Length(Path\_between(tst, x, y)) \text{ if } x, y \in A \wedge x.t_e < y.t_s) \vee (Length(Path\_between(tst, x, y)) \text{ if } x, y \in A \wedge y.t_e < x.t_s)$ (IV.97)
(III.111)	$f_{TST\_Distance\_TST}(tst_1, tst_2) := distance(Itinerary(tst_1), Itinerary(tst_2))$ (IV.98)
(III.112)	$f_{TST\_Distance\_Point}(tst, p) := distance(Itinerary(tst), p)$ (IV.99)
(III.113)	$f_{TST\_Distance\_Polyline}(tst, l) := distance(Itinerary(tst), l)$ (IV.100)
(III.114)	$f_{TST\_Distance\_Region}(tst, r) := distance(Itinerary(tst), r)$ (IV.101)

Tableau IV.16 – Sémantique des opérations de calcul spatial

La longueur partielle d'une trajectoire spatio-temporelle  $tst \in D_{tst}$  est représentée par la longueur du chemin entre deux activités. Cette longueur est aussi calculée en appliquant l'opérateur  $Length$  au chemin entre les deux activités. Ceci est traduit mathématiquement par la fonction (IV.97). Le chemin en question est récupéré à partir de l'opérateur  $Path\_Between$ , dont la sémantique est défini précédemment par la fonction (IV.74).

La distance entre une valeur de trajectoire  $tst$  appartenant à  $D_{tst}$  et une entité géométrique est déterminée par le calcul de la distance spatiale entre l'entité en question et le polygone, résultat de la projection de  $tst$  dans le domaine spatial. Les fonctions (IV.99), (IV.100), (IV.87) et (IV.98) expriment la sémantique de ces opérations.

### 3.4.5 Opérations de recherche spatiale

Les opérations de recherche d'événements, selon un critère spatial, prennent en argument une valeur de trajectoire,  $tst \in D_{tst}$ , et une entité géométrique et retournent un ensemble d'activités ou de déplacements de la  $tst$  qui se trouvent spatialement à cette entité géométrique. La sémantique de l'ensemble des opérations de ce type est donnée par le tableau IV.17.

Lorsqu'il s'agit par exemple d'une entité géométrique  $p$ , de type  $Point$ , l'opération  $Activities\_At\_Point(tst, p)$  retourne les activités dont la localisation est spatialement égale à  $p$ . Ceci est traduit par la fonction (IV.102). L'opération  $Trips\_At\_Point(tst, p)$  retourne, quant à elle, les déplacements qui ont traversés le point  $p$ ; c'est à dire tels que  $p$  est à l'intérieur (i.e *inside*) de leur chemin spatial  $path$ . Ceci est traduit par la fonction (IV.103).

Lorsqu'il s'agit d'une polygone,  $l \in Polyline$ , l'opération  $Activities\_At\_Polyline(tst, l)$ , retourne les activités de la trajectoire  $tst$  dont les localisations appartiennent à  $l$ . L'opération

## Chapitre IV. Algèbre de définition et de manipulation du TST

---

$Trips\_At\_Polyline(tst, l)$  retourne, quant à elle, les déplacements de la trajectoire dont le chemin appartient au polygone  $l$ . La sémantique de ces deux opérations est exprimée par les fonctions (IV.105) et (IV.104).

Signature	Fonction d'interprétation
(III.72)	$f_{Activities\_At\_Point}(tst, p) := \{a \in A   a.l = p\}$ (IV.102)
(III.75)	$f_{Trips\_At\_Point}(tst, p) := \{d \in D   p \text{ inside } d.path\}$ (IV.103)
(III.78)	$f_{Activities\_At\_Polyline}(tst, l) := \{a \in A   a.l \text{ inside } l\}$ (IV.104)
(III.81)	$f_{Trips\_At\_Polyline}(tst, l) := \{d \in D   d.path \text{ inside } l\}$ (IV.105)
(III.87)	$f_{Trips\_At\_Region}(tst, r) := \{d \in D   d.path \text{ inside } r\}$ (IV.106)
(III.84)	$f_{Activities\_At\_Region}(tst, r) := \{a \in A   a.l \text{ inside } r\}$ (IV.107)

**Tableau IV.17** – Sémantique des opérations de recherche spatiale retournant une liste d'événements

Nous avons aussi proposé un groupe d'opérations de recherche spatiale qui, étant donné une trajectoire, retourne le premier ou le dernier événement vérifiant un certain critère spatial. La figure (IV.4) illustre un exemple d'utilisation et de visualisation de certaines de ces opérations sur un TST dans un espace tri-dimensionnel. Nous utilisons la relation d'ordre au sein du type TST afin d'exprimer la sémantique de ce groupe d'opérations dans les fonctions se trouvant dans le tableau IV.18.

Signature	Fonction d'interprétation
(III.73)	$f_{FirstActivity\_At\_Point}(tst, p) := a_i \in A : (a_i.l = p) \wedge (i \nexists j \neq i, 1 \leq j \leq n \text{ such that } a_j.l = p \Rightarrow i < j)$ (IV.108)
(III.76)	$f_{FirstTrip\_At\_Point}(tst, p) := d_i \in D : (p \text{ inside } d_i.path) \wedge (i \nexists j \neq i, 1 \leq j \leq n \text{ such that } p \text{ inside } d_i.path \Rightarrow i < j)$ (IV.109)
(III.79)	$f_{FirstActivity\_At\_Polyline}(tst, l) := a_i \in A : (a.l \text{ inside } l \wedge (i \nexists j \neq i, 1 \leq j \leq n \text{ such that } a.l \text{ inside } l \Rightarrow i < j))$ (IV.110)
(III.81)	$f_{FirstTrip\_At\_Polyline}(tst, l) := d_i \in D : (d_i.path \text{ inside } l) \wedge (i \nexists j \neq i, 1 \leq j \leq n \text{ such that } d_i.path \text{ inside } l \Rightarrow i < j)$ (IV.111)
(III.85)	$f_{FirstActivity\_At\_Region}(tst, r) := a_i \in A : (a.l \text{ inside } r) \wedge (i \nexists j \neq i, 1 \leq j \leq n \text{ such that } a.l \text{ inside } r \Rightarrow i < j)$ (IV.112)
(III.88)	$f_{FirstTrip\_At\_Region}(tst, r) := d_i \in D : (d_i.path \text{ inside } r) \wedge (i \nexists j \neq i, 1 \leq j \leq n \text{ such that } d_i.path \text{ inside } r \Rightarrow i < j)$ (IV.113)
(III.74)	$f_{LastActivity\_At\_Point}(tst, p) := a_i \in A : (a_i.l = p) \wedge (i \nexists j \neq i, 1 \leq j \leq n \text{ such that } a_j.l = p \Rightarrow i > j)$ (IV.115)
(III.77)	$f_{LastTrip\_At\_Point}(tst, p) := d_i \in D : (p \text{ inside } d_i.path) \wedge (i \nexists j \neq i, 1 \leq j \leq n \text{ such that } p \text{ inside } d_i.path \Rightarrow i > j)$ (IV.116)
(III.80)	$f_{LastActivity\_At\_Polyline}(tst, l) := a_i \in A : (a.l \text{ inside } l) \wedge (i \nexists j \neq i, 1 \leq j \leq n \text{ such that } a.l \text{ inside } l \Rightarrow i > j)$ (IV.117)
(III.83)	$f_{LastTrip\_At\_Polyline}(tst, l) := d_i \in D : (d_i.path \text{ inside } l) \wedge (i \nexists j \neq i, 1 \leq j \leq n \text{ such that } d_i.path \text{ inside } l \Rightarrow i > j)$ (IV.118)

(III.86)	$f_{LastActivity\_At\_Region}(tst, r) := a_i \in A : (a.l \text{ inside } r) \wedge (if \exists j \neq i, 1 \leq j \leq n \text{ such that } a.l \text{ inside } r)$ $\Rightarrow i > j$	(IV.119)
(III.89)	$f_{LastTrip\_At\_Region}(tst, r) := d_i \in D : (d_i.path \text{ inside } r) \wedge (if \exists j \neq i, 1 \leq j \leq n \text{ such that } d_i.path \text{ inside } r)$ $\Rightarrow i > j$	(IV.120)

**Tableau IV.18** – Sémantique des opérations de recherche sémantique retournant un événement

### 3.4.6 Opérations de similarité spatiale

Étant donné deux valeur de TST,  $tst_1$  et  $tst_2$ , ces trajectoires sont spatialement similaires si leur itinéraires spatiales sont identiques et donc les polygones résultats de l'opération *Itinerary* sont égales spatialement. Ceci est traduit par la fonction d'interprétation (IV.121)).

Signature	Fonction d'interprétation
(III.115)	$f_{Spatial\_Similarity}(tst_1, tst_2) := Itinerary(tst_1) Equal Itinerary(tst_2)$ (IV.121)

Tableau IV.19 – Sémantique de l'opération de similarité spatiale

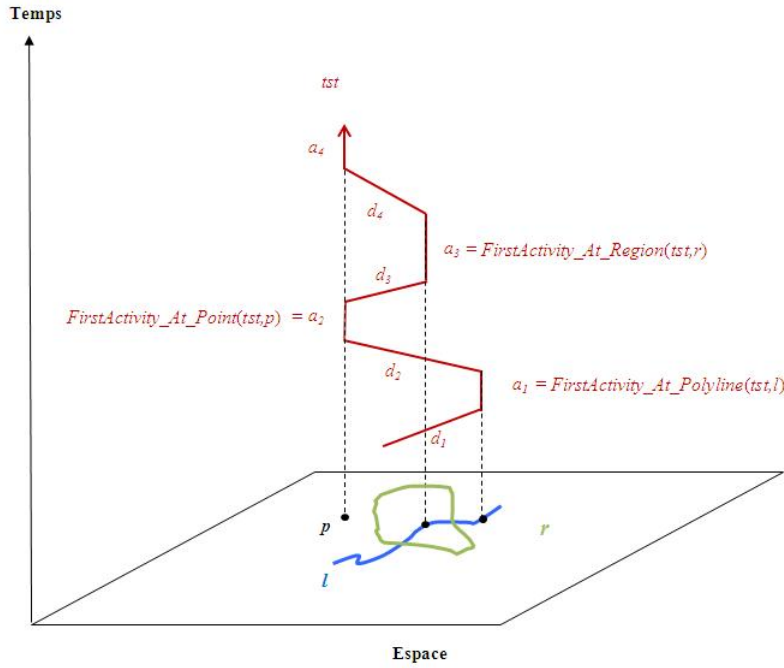


Figure IV.4 – Exemples d'opérations de recherche spatiale dans un TST

## 3.5 Opérations temporelles

### 3.5.1 Opérations de relations temporelles

Dans cette partie, nous assimilons une trajectoire spatio-temporelle à son intervalle temporel de définition  $I_{tst} = [d_1.t_s, a_n.t_e]$ . L'interprétation des opérations de topologie temporelles proposées est analogue à l'interprétation de la sémantique des opérations d'Allen [23]. Dans notre modèle, les opérations vérifient les relations temporelles entre, d'une part, un instant  $t \in Time$



## Chapitre IV. Algèbre de définition et de manipulation du TST

---

ou un intervalle  $I \in Interval$ , et d'autre part l'intervalle  $I_{tst}$ . À titre d'exemple, selon Allen, pour qu'un instant  $t$  soit contenu dans un intervalle de temps fermé  $I = [b_{inf}, b_{sup}]$ , il faut que  $t$  soit strictement supérieur à  $b_{inf}$  et strictement inférieur à  $b_{sup}$ . De même, pour qu'un instant  $t$  soit contenu dans une trajectoire  $tst \in D_{tst}$ , il faut que  $t$  soit supérieur à  $d_1.t_s$  et inférieur à  $a_n.t_e$ . Le même raisonnement est appliqué pour les autres opérations de topologie temporelle entre, d'une part, une trajectoire  $tst$  et un instant de temps, et d'autre part, une trajectoire  $tst$  et un intervalle de temps. La sémantique de la totalité des opérations de topologie temporelle est résumée dans le tableau IV.20.

Signature	Fonction d'interprétation
(III.116)	$f_{TST\_Before\_Time}(tst, t) := a_n.t_e < t$ (IV.122)
(III.117)	$f_{TST\_After\_Time}(tst, t) := t < d_1.t_s$ (IV.123)
(III.118)	$f_{TST\_EndedBy\_Time}(tst, t) := a_n.t_e = t$ (IV.124)
(III.120)	$f_{TST\_BegunBy\_Time}(tst, t) := d_1.t_s = t$ (IV.125)
(III.119)	$f_{TST\_Contains\_Time}(tst, t) := d_1.t_s < t \wedge t < a_n.t_e$ (IV.126)
(III.121)	$f_{TST\_Before\_Interval}(tst, I) := tst.a_n.t_e < I.t_s$ (IV.127)
(III.122)	$f_{TST\_After\_Interval}(tst, I) := tst.d_1.t_s > I.t_e$ (IV.128)
(III.123)	$f_{TST\_Meets\_Interval}(tst, I) := tst.a_n.t_e = I.t_s$ (IV.129)
(III.124)	$f_{TST\_Overlaps\_Interval}(tst, I) := (tst.d_1.t_s < I.t_s) \wedge (tst.a_n.t_e < I.t_e) \wedge (tst.a_n.t_e > I.t_s)$ (IV.130)
(III.125)	$f_{TST\_Begins\_Interval}(tst, I) := (tst.d_1.t_s = I.t_s) \wedge (tst.a_n.t_e < I.t_e)$ (IV.131)
(III.126)	$f_{TST\_BegunBy\_Interval}(tst, I) := (tst.d_1.t_s = I.t_s) \wedge (tst.a_n.t_e > I.t_e)$ (IV.132)
(III.127)	$f_{TST\_During\_Interval}(tst, I) := (tst.d_1.t_s > I.t_s) \wedge (tst.a_n.t_e < I.t_e)$ (IV.133)
(III.128)	$f_{TST\_Equals\_Interval}(tst, I) := (tst.d_1.t_s = I.t_s) \wedge (tst.a_n.t_e = I.t_e)$ (IV.134)
(III.129)	$f_{TST\_OverlappedBy\_Interval}(tst, I) := (tst.d_1.t_s > I.t_s) \wedge (tst.a_n.t_e > I.t_e)$ $\wedge (tst.d_1.t_s < I.t_e)$ (IV.135)
(III.130)	$f_{TST\_Ends\_Interval}(tst, I) := (tst.d_1.t_s > I.t_s) \wedge (tst.a_n.t_e = I.t_e)$ (IV.136)
(III.131)	$f_{TST\_EndedBy\_Interval}(tst, I) := (tst.d_1.t_s < I.t_s) \wedge (tst.a_n.t_e = I.t_e)$ (IV.137)
(III.132)	$f_{TST\_MetBy\_Interval}(tst, I) := tst.d_1.t_s = I.t_e$ (IV.138)
(III.140)	$f_{TST\_TemporalEquals\_TST}(tst1, tst2) := (tst1.d_1.t_s = tst2.d_1.t_s) \wedge (tst1.a_n.t_e =$ $tst2.a_n.t_e)$ (IV.139)

(III.133)	$f_{TST\_Before\_TST}(tst1, tst2) := tst1.a_n.t_e < tst2.d_1.t_s$	(IV.140)
(III.134)	$f_{TST\_After\_TST}(tst1, tst2) := tst2.a_n.t_e < tst1.d_1.t_s$	(IV.141)
(III.135)	$f_{TST\_Meets\_TST}(tst1, tst2) := tst1.a_n.t_e = tst2.d_1.t_s$	(IV.142)
(III.136)	$f_{TST\_Overlaps\_TST}(tst1, tst2) := (tst1.d_1.t_s < tst2.d_1.t_s) \wedge (tst1.a_n.t_e < tst2.a_n.t_e)$ $\wedge (tst1.a_n.t_e > tst2.d_1.t_s)$	(IV.143)
(III.137)	$f_{TST\_Before\_TST}(tst1, tst2) := tst1.a_n.t_e < tst2.d_1.t_s$	(IV.144)
(III.139)	$f_{TST\_During\_TST}(tst1, tst2) := (tst1.d_1.t_s > tst2.d_1.t_s) \wedge (tst1.a_n.t_e < tst2.a_n.t_e)$	(IV.145)
(III.141)	$f_{TST\_OverlappedBy\_TST}(tst1, tst2) := (tst1.d_1.t_s > tst2.d_1.t_s) \wedge (tst1.a_n.t_e > tst2.a_n.t_e)$ $\wedge (tst1.d_1.t_s < tst2.a_n.t_e)$	(IV.146)
(III.142)	$f_{TST\_Ends\_TST}(tst1, tst2) := (tst1.d_1.t_s > tst2.d_1.t_s) \wedge (tst1.a_n.t_e = tst2.a_n.t_e)$	(IV.147)
(III.143)	$f_{TST\_EndedBy\_TST}(tst1, tst2) := (tst1.d_1.t_s < tst2.d_1.t_s) \wedge (tst1.a_n.t_e = tst2.a_n.t_e)$	(IV.148)

Tableau IV.20 – Sémantique des opération de topologie temporelle

### 3.5.2 Opérations de calcul des durées

Le calcul de la durée d'une trajectoire  $tst \in D_{tst}$  de taille  $n$  se fait en calculant la durée de temps entre l'instant de début et l'instant de fin de la trajectoire. Ceci s'exprime par la fonction (IV.149).

La sémantique de la fonction qui calcule la durée totale qu'un objet mobile a passé en activité s'exprime en calculant la somme des durées de toutes les activités de la trajectoire de cet objet. Le même raisonnement est appliqué au calcul des durées des déplacements d'une trajectoire. La sémantique de ces opérations est illustrée par les fonctions (IV.150) et (IV.151).

### 3.5.3 Opérations de restriction temporelle

Étant donné une trajectoire  $tst \in D_{tst}$  et un instant de temps  $t \in Time$ , la restriction temporelle de  $tst$  à l'instant  $t$  est une sous-trajectoire composée d'un couple d'activité/déplacement,  $(a_i, d_i)$ . L'activité et le déplacement en question doivent appartenir à la trajectoire  $tst$  et apparaître dans le chaînage dans le même ordre  $i$ . L'instant de temps  $t$  doit être contenu, soit dans

Signature	Fonction d'interprétation
(III.145)	$f_{Duration}(tst) := a_n.t_e - d_1.t_s$ (IV.149)
(III.146)	$f_{Activities\_Duration}(tst) := \sum_{i=1}^n (a_i.t_e - a_i.t_s)$ (IV.150)
(III.147)	$f_{Trips\_Duration}(tst) := \sum_{i=1}^n (d_i.t_e - d_i.t_s)$ (IV.151)

**Tableau IV.21** – Sémantique des opérations de calcul des durées

l'intervalle de temps pendant lequel s'est déroulé  $a_i$ , soit dans l'intervalle de temps pendant lequel s'est déroulé  $d_i$ . Ceci est exprimé par la fonction (IV.152).

Étant donné une trajectoire  $tst \in D_{tst}$  et un intervalle  $I = [b_{inf}, b_{sup}] \in Interval$ , la restriction temporelle de la trajectoire  $tst$  à l'intervalle  $I$  est une sous-trajectoire de  $tst$  complètement incluse temporellement dans  $I$ . Ceci s'exprime par un couple, que nous nommons ici par  $(A', D')$ , composé d'un ensemble ordonné,  $A'$ , d'activités tel que  $A' \subset A$  et d'un ensemble temporellement ordonné,  $D'$ , de déplacements, tel que  $D' \subset D$ . Pour que cette sous-trajectoire soit incluse dans  $I$ , il faut que le temps de début du premier déplacement de  $D'$  soit supérieur ou égal à la borne inférieure de  $I$  et il faut que l'instant de fin de la dernière activité de  $A'$  soit inférieur ou égal à la borne supérieur de  $I$ . Cette sémantique est exprimée mathématiquement par la fonction (IV.153).

Signature	Fonction d'interprétation
(III.148)	$f_{At\_Time}(tst, t) := (a_i \in A, d_i \in D) \text{ if } \exists 1 \leq i \leq n : (t \geq d_i.t_s) \wedge (t \leq a_i.t_e)$ (IV.152)
(III.149)	$f_{At\_Interval}(tst, I) := (\{a_{min}, a_{min+1}, \dots, a_{max}\} \subset A, \{d_{min}, d_{min+1}, \dots, d_{max}\} \subset D)$ if $\exists (1 \leq min \leq n \wedge 1 \leq max \leq n) : (min \leq max) \wedge (d_{min}.t_s \geq I.b_{inf})$ $\wedge (a_{max}.t_e \leq I.b_{sup})$ (IV.153)

**Tableau IV.22** – Sémantique des opérations de restriction temporelle

### 3.5.4 Opérations de projection dans le domaine temporel

Les opérations de projection temporelle d'une trajectoire  $tst \in D_{tst}$  à des instants de temps sont *Times*, *First\_Time* et *End\_Time*. La sémantique de la première s'exprime par la fonction (IV.154)) qui retourne les instants de début et de fin de toutes les activités et de tous les

déplacements la trajectoire  $tst$ .

Les sémantiques de la seconde et de la troisième opération s'exprime par les fonctions, (IV.155) et (IV.156), qui retournent, respectivement, le temps de début du premier déplacement et le temps de fin de la dernière activité.

La projection temporelle d'une trajectoire à un intervalle de temps s'exprime par la fonction (IV.157) qui retourne l'intervalle de définition de la trajectoire, à savoir  $I_{tst} = [d_1.t_s, a_n.t_e]$ .

Signature	Fonction d'interprétation
(III.151)	$f_{Times}(tst) := \{t \in Time : (t = d.t_s) \vee (t = d.t_e) \vee (t = a.t_s) \vee (t = a.t_e)\}$ (IV.154)
(III.152)	$f_{First\_Time}(tst) := d_1.t_s \in Time$ (IV.155)
(III.153)	$f_{Last\_Time}(tst) := a_n.t_e \in Time$ (IV.156)
(III.150)	$f_{Temporal\_Length}(tst) := I \in Interval : I = [d_1.t_s, a_n.t_e]$ (IV.157)

Tableau IV.23 – Sémantique des opérations de projection dans le domaine temporel

### 3.5.5 Opérations de recherche temporelle

Les opérations de recherche temporelle retournent des activités ou des déplacements qui se sont produits à un instant de temps ou pendant un intervalle de temps donné. En effet, étant donné une valeur de trajectoire  $tst \in D_{tst}$  et un instant de temps  $t \in Time$ , la sémantique de l'opération  $Activity\_At\_Time$  est exprimée par la fonction (IV.158) qui retourne une unique activité,  $a_i \in A$ , si elle existe. L'activité  $a_i$  doit temporellement contenir, commencer ou se terminer par l'instant  $t$ . Le même raisonnement est appliqué pour la sémantique de l'opération  $Trip\_At\_Time$  (voir la fonction (IV.159)).

Étant donné, maintenant, un intervalle  $I = [b_{inf}, b_{sup}] \in Interval$  et une valeur de trajectoire  $tst \in D_{tst}$ , durant l'intervalle  $I$ , plus d'un événement peuvent se produire. Ainsi, l'interprétation de l'opération  $Activities\_At\_Interval(tst, I)$  s'exprime par la fonction (IV.160) qui retourne un ensemble d'activités  $a_j \in A$ , tel que l'intervalle de déroulement de ces activités a lieu pendant l'intervalle  $I$ . Le même raisonnement est appliqué pour la sémantique de l'opération  $Trips\_At\_Interval$  (voir la fonction (IV.161)).

### 3.5.6 Opérations de similarité temporelle

Pour que deux valeurs de trajectoires  $tst_1$  et  $tst_2$  soient temporellement similaires, il faut impérativement, qu'elles aient le même nombre d'éléments (activités et déplacements). De plus,

Signature	Fonction d'interprétation
(III.154)	$f_{Activity\_At\_Time}(tst, t) := a_i \in A \text{ if } \exists 1 \leq i \leq n : t \geq a_i.t_s \wedge t \leq a_i.t_e$ (IV.158)
(III.155)	$f_{Trip\_At\_Time}(tst, t) := d_i \in D, \text{ if } \exists 1 \leq i \leq n : t \geq d_i.t_s \wedge t \leq d_i.t_e$ (IV.159)
(III.156)	$f_{Activities\_At\_Interval}(tst, I) := \{a_i \in A   1 \leq i \leq n \wedge I.b_{inf} \leq a_i.t_s \wedge I.b_{sup} \geq a_i.t_e\}$ (IV.160)
(III.157)	$f_{Trips\_At\_Interval}(tst, I) := \{d_i \in D   1 \leq i \leq n \wedge I.b_{inf} \leq d_i.t_s \wedge I.b_{sup} \geq d_i.t_e\}$ (IV.161)

Tableau IV.24 – Sémantique des opérations de recherche temporelle

il faut vérifier l'égalité temporelle entre les instants de début et de fin des activités et des déplacements de même indice dans les ensembles  $A$  et  $D$  des deux trajectoires  $tst_1$  et de  $tst_2$ . Ceci est exprimé par la fonction (IV.162).

Signature	Fonction d'interprétation
(III.115)	$f_{Temporal\_Similarity}(tst_1, tst_2) := (tst_1.D.d_1.t_s = tst_2.D.d_2.t_s) \wedge (\forall 1 \leq i \leq n, \\ tst_1.A.a_i.t_s = tst_2.A.a_i.t_s \wedge tst_1.A.a_i.t_e = tst_2.A.a_i.t_e)$ (IV.162)

Tableau IV.25 – Sémantique des opérations de similarité temporelle

## 3.6 Opérations spatio-temporelles

### 3.6.1 Opérations ensemblistes

Nous proposons dans cette partie la sémantique des opérations ensemblistes dont la signature a été présentée dans le chapitre précédent. Le tableau IV.26 récapitule toutes les fonctions d'interprétation associées aux opérations ensemblistes.

**Opération de test d'égalité** L'opération de test d'égalité spatio-temporelle *Equals* évalue si deux trajectoires comportent exactement les mêmes événements, c'est-à-dire des événements qui se sont déroulés aux mêmes localisations, aux mêmes instants, tout en traduisant la même sémantique. La fonction d'interprétation (IV.163) de ce prédicat illustre cette sémantique.

**Opération de test d'inclusion** Pour qu'une trajectoire  $tst_1 \in TST$  soit incluse dans une trajectoire  $tst_2 \in TST$ , il suffit que l'ensemble des activités et l'ensemble des déplacements de  $tst_1$  soient inclus, respectivement, dans l'ensemble des activités et dans l'ensemble des déplacements de  $tst_2$ . L'inclusion doit être stricte pour éviter le cas de l'égalité, qui peut être vérifié par l'opération *Equals*. La sémantique de l'opération d'inclusion entre TST *Includes* traduit par la fonction d'interprétation (IV.165).

**Opération de test d'union** Dans le modèle syntaxique du type TST, l'union de deux trajectoires, traduit par la fonction *Union*, est une trajectoire dont les éléments respectent les règles de chaînage et de succession temporelle et spatiale. Or, l'opération d'union, dans le sens ensembliste, de deux ensembles E et F retourne un ensemble qui contient des éléments qui appartiennent soit à E soit à F :  $E \cup F = \{x \in E \vee x \in F\}$ . Si nous appliquons cette opération à des trajectoires dont les activités se chevauchent, le résultat serait l'ensemble de tous les événements de  $tst_1$  et de  $tst_2$ . Or, ce résultat ne peut en aucun cas former un TST valide, vu qu'il ne vérifie pas les conditions de chaînage au sein d'une trajectoire spatio-temporelle et ceci à cause du chevauchement des activités au sein des TST.

Comme solution à cette problématique, nous distinguons différents cas de figure de trajectoires pour lesquels l'opération d'union sur le type TST pourrait être appliquée (voir figure IV.5), à savoir :

1. Le cas où les deux trajectoires sont égales (c'est à dire  $Equals(tst_1, tst_2)$  est vrai). Dans ce cas, le résultat de l'union donne l'une des deux trajectoires.
2. Le cas où l'une des trajectoires est incluse dans l'autre. Autrement dit :
  - si  $Includes(tst_1, tst_2)$  est vrai alors  $Union(tst_1, tst_2) = Union(tst_2, tst_1)$  retourne  $tst_2$ ,
  - si  $Includes(tst_2, tst_1)$  est vrai alors  $Union(tst_1, tst_2) = Union(tst_2, tst_1)$  retourne  $tst_1$ .
3. Le cas où les deux trajectoires se succèdent temporellement et spatialement. Autrement dit, les intervalles temporels de définition des deux trajectoires doivent se rencontrer (à une unité de temps près) et les localisations  $a_n.l$  de l'une et  $d_1.l_s$  de l'autre doivent coïncider. C'est le cas simplifié traité dans l'article [3].
4. Le cas où les deux trajectoires ont un ensemble d'activités et de déplacements successifs en commun et que cet ensemble se situe respectivement au début de l'une des trajectoires et à la fin de l'autre.

Schématiquement parlant, dans les deux derniers cas, c'est comme si on "collait" ou on concaténait dans un espace 3D (tel que proposé par la Time Geography) un TST à l'extrémité

## Chapitre IV. Algèbre de définition et de manipulation du TST

d'un deuxième TST pour en créer un troisième. Si les deux TST ont un "bout" d'extrémité en commun, les parties communes seront superposées l'une sur l'autre.

La précedence temporelle va dicter le sens de la concaténation. Autrement dit, le traitement de l'opération  $Union(tst_1, tst_2)$  sera analogue à la concaténation de :

- $tst_1$  à la fin de  $tst_2$  si  $tst_2$  précède temporellement  $tst_1$ ,
- $tst_2$  à la fin de  $tst_1$  si  $tst_1$  précède temporellement  $tst_2$ .

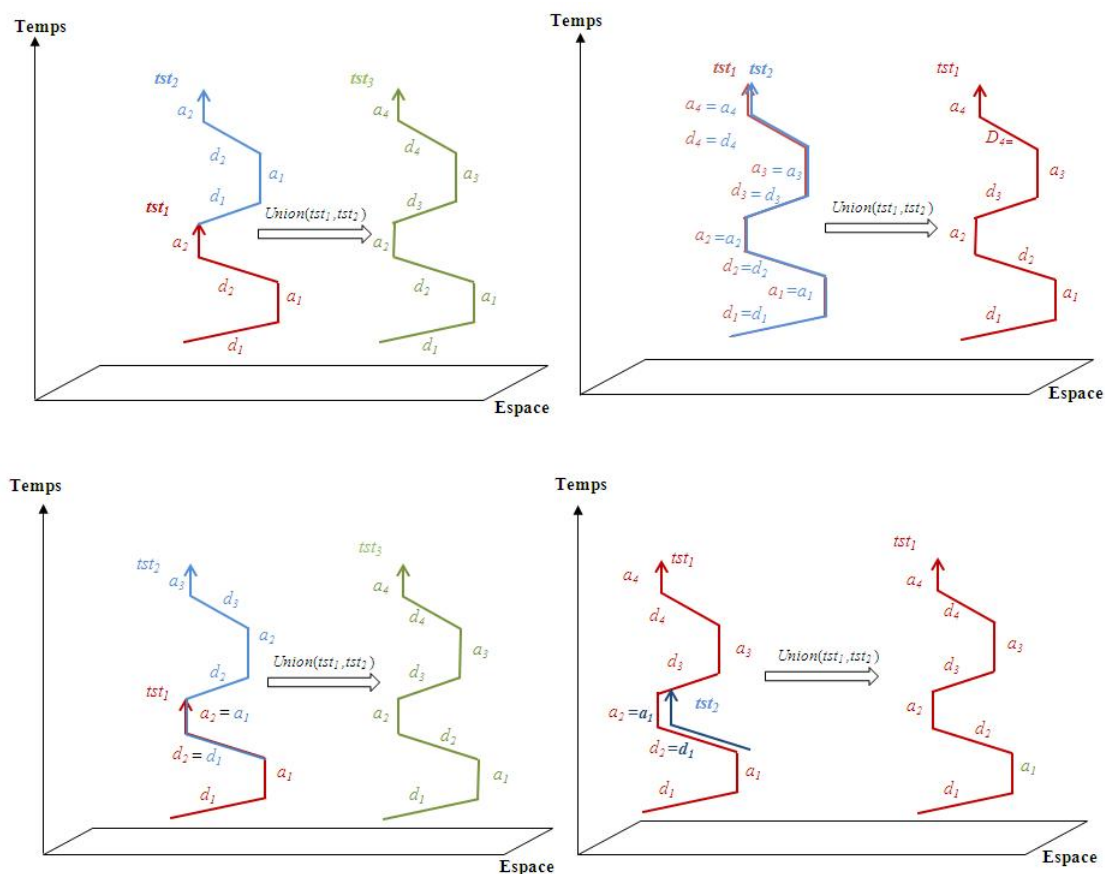


Figure IV.5 – Les différents cas possibles d'Union de deux TST

La fonction d'interprétation de l'opération  $Union$  que nous proposons est illustrée par dans l'équation (IV.170).

**Opérations de test d'intersection** Deux entités  $tst_1$  et  $tst_2$  de type TST s'intersectent si elles ont en commun, au moins, un déplacement ou une activité. Ceci est traduit par le prédicat  $Intersects$ , dont la fonction d'interprétation (IV.164).

**Opérations d'intersection** Nous avons aussi proposé dans le modèle syntaxique du type TST trois autres opérations (non booléennes) d'intersection. La première, *Interesction* (voir l'équation (III.163) du chapitre précédent), appliquée entre deux entités de type TST, retourne une trajectoire de type *TST* qui forme l'intersection de ces deux TST. Vu qu'il peut exister plusieurs sous-trajectoires communes entre des TST, nous décidons que l'opération *Interesction* ne renvoie que la première TST d'intersection. L'utilisateur pourra retrouver les autres trajectoires d'intersection en ré-appliquant cette opération sur des sous-trajectoires restreintes.

Mathématiquement parlant, l'opération *Interesction*, retournera une paire, représentant une sous-trajectoire, *Sous\_TST*, composée de deux ensembles ordonnés d'activités et de déplacements. Pour que l'inclusion soit vérifiée, chacun de ces deux ensembles ordonnés doit être inclus dans les deux TST en arguments. Pour que la valeur *Sous\_TST* soit la première trajectoire d'intersection dans le chaînage, il ne doit pas exister de sous-trajectoires incluses dans les trajectoires en arguments et telles qu'elles précèdent temporellement la valeur *Sous\_TST*.

Étant donné deux entités  $tst_1$  et  $tst_2 \in TST$ , de cardinalités respectives  $n_1$  et  $n_2 \in \mathbb{N}$ , la sémantique de l'opération *Interesction*( $tst_1, tst_2$ ) est décrite par la fonction (IV.166).

La seconde opération d'intersection, *Activities\_Intersection*, retourne l'ensemble des activités communes entre deux entités  $tst_1$  et  $tst_2 \in TST$ . La fonction d'interprétation de cette opération est décrite dans l'équation (IV.167).

Enfin, la dernière opération d'intersection, *Trips\_Intersection*, retourne l'ensemble des déplacements commune entre deux entités  $tst_1$  et  $tst_2 \in TST$ . La fonction d'interprétation de cette opération est décrite dans l'équation (IV.168).

**Opération de différence** La différence entre deux ensembles E et F est un ensemble qui contient des éléments qui appartiennent à E mais pas à F :  $E \setminus F = \{x \in E | x \notin F\}$ . Comme pour l'opération d'inclusion, nous avons proposé trois opérations de différence entre des entités de type TST.

La première opération, appelée *Difference*, retourne une trajectoire de type TST. En effet, étant donné deux entités  $tst_1$  et  $tst_2 \in TST$ , le TST résultat est une sous-trajectoire incluse à (forme un fragment de)  $tst_1$  mais non incluse à  $tst_2$ . Vu qu'il peut exister plusieurs sous-trajectoires appartenant à  $tst_2$  et non à  $tst_1$ , l'opération *Difference* retournera la première (dans le chaînage des événements d'un TST) sous-trajectoire vérifiant cette condition. Ceci est traduit sémantiquement par la fonction (IV.169).

La deuxième opération de différence, *Trips\_Difference*, retourne l'ensemble des déplacements appartenant à la trajectoires  $tst_1$  mais pas à la trajectoire  $tst_2$ . La fonction d'interprétation de cette opération est décrite par la fonction (IV.172).



La seconde opération de différence *Activities\_Difference*, retourne l'ensemble des activités appartenant à la trajectoires  $tst_1$  mais pas à la trajectoire  $tst_2 \in TST$ . La fonction d'interprétation de cette opération est décrite par la fonction (IV.171).

### 3.6.2 Opérations de voisinage spatio-temporel

Étant donné une valeur de trajectoire  $tst \in TST$ , une entité spatiale, que nous allons nommer  $g$ , un seuil de distance  $s \in Real$  et un instant de temps  $t \in time$ , la vérification du voisinage spatio-temporel, illustrée par les fonctions (IV.174), (IV.175) et (IV.176), entre la valeur  $tst$  et l'entité  $g$  s'exprime comme suit :

- Il faut que la trajectoire  $tst$  soit définie à l'instant  $t$ . L'utilisation de l'opération  $existe\_tst : TST \times Time \rightarrow Boolean$  est alors nécessaire.
- Si à l'instant  $t$  il y a occurrence d'une activité au sein de la trajectoire, il faut tester alors le voisinage entre la localisation spatiale de l'activité en question (e.g  $a.l$ ) et l'entité spatiale.
- Si à l'instant  $t$  il y a occurrence d'un déplacement au sein de la trajectoire, alors on ne pourra pas avoir l'emplacement exact de l'objet mobile vu le caractère discontinu de notre modélisation. Nous proposons néanmoins une solution qui fournit une approximation du voisinage en testant sur le voisinage entre l'itinéraire du déplacement (e.g  $d.path$ ) et l'entité spatiale. Cette solution indique qu'au cours du déplacement qui a eu lieu à l'instant  $t$ , la trajectoire a été voisine de l'entité spatiale  $g$  selon le seuil  $s$ .

Étant donné maintenant deux valeurs de trajectoires  $tst1$  et  $tst2 \in TST$ , un seuil de distance  $s \in Real$  et un instant de temps  $t \in time$ , la vérification du voisinage spatio-temporel, illustrée par la fonction (IV.173) entre les trajectoires  $tst1$  et  $tst2$  s'exprime comme suit :

- Il faut que les trajectoires  $tst1$  et  $tst2$  soient définies à l'instant  $t$ . L'utilisation de l'opération  $existe\_tst : TST \times Time \rightarrow Boolean$  est alors nécessaire.
- Si à l'instant  $t$  il y a occurrence d'activités au sein des deux trajectoires, il faut tester alors le voisinage entre les localisations spatiales des deux activités.
- Si à l'instant  $t$  il y a occurrence de déplacements au sein des deux trajectoires, alors on ne pourra pas avoir l'emplacement exact de l'objet mobile vu le caractère discontinu de notre modélisation. Comme solution approximative, il faut tester alors le voisinage entre les itinéraires spatiaux des deux déplacements.
- Si à l'instant  $t$  il y a occurrence d'un déplacement  $d$  au sein d'une des trajectoires et d'une activité  $a$  au sein de l'autre, alors tester le voisinage spatio-temporel entre les deux TST revient à tester le voisinage entre l'itinéraire  $d.path$  de  $d$  et la localisation  $a.l$  de  $a$ .

Signature	Fonction d'interprétation
(III.159)	$f_{Equals}(tst_1, tst_2) := (tst_1.A = tst_2.A) \wedge (tst_1.D = tst_2.D)$ (IV.163)
(III.161)	$f_{Intersects}(tst_1, tst_2) := (tst_1.A \cap tst_2.A \neq \emptyset) \vee (tst_1.D \cap tst_2.D \neq \emptyset)$ (IV.164)
(III.160)	$f_{Includes}(tst_1, tst_2) := (tst_1.A \subseteq tst_2.A) \wedge (tst_1.D \subseteq tst_2.D)$ (IV.165)
(III.163)	$f_{Intersection}(tst_1, tst_2) := \left\{ \begin{array}{l} \text{Sous\_TST} = \{a_i, a_{i+1}, \dots, a_j\}, \\ \{d_i, d_{i+1}, \dots, d_j\} \end{array} \right.$ <p style="text-align: center;">                     if <math>\exists(1 \leq i \leq \min(n_1, n_2) \wedge 1 \leq j \leq \min(n_1, n_2)) :</math>  <math>\text{Sous\_TST} \subset D_{TST} \wedge \text{Includes}(\text{Sous\_TST}, tst_1) \wedge</math>  <math>\text{Includes}(\text{Sous\_TST}, tst_2) \wedge (\exists!H : (H \subset D_{TST} \wedge</math> (IV.166)  <math>\wedge \text{Includes}(H, tst_1) \wedge \text{Includes}(H, tst_2) \wedge</math>  <math>\text{TST\_Before\_TST}(H, \text{Sous\_TST}))</math>                      else  <math>\emptyset</math> </p>
(III.164)	$f_{Activities\_Intersection}(tst_1, tst_2) := \{a \in D_a \mid a \in tst_1.A \wedge a \in tst_2.A\}$ (IV.167)
(III.165)	$f_{Trips\_Intersection}(tst_1, tst_2) := \{d \in D_d \mid d \in tst_1.D \wedge d \in tst_2.D\}$ (IV.168)
(III.166)	$f_{Difference}(tst_1, tst_2) := \left\{ \begin{array}{l} \text{Sous\_TST} = \{a_i, a_{i+1}, \dots, a_j\}, \\ \{d_i, d_{i+1}, \dots, d_j\} \end{array} \right.$ <p style="text-align: center;">                     if <math>\exists(1 \leq i \leq n_1 \wedge 1 \leq j \leq n_1) : \text{Sous\_TST} \subset D_{TST} \wedge</math>  <math>\text{Includes}(\text{Sous\_TST}, tst_1) = \text{true} \wedge \text{Includes}(\text{Sous\_TST}, tst_2) = \text{false} \wedge (\exists!H : (H \subset D_{TST} \wedge</math> (IV.169)  <math>\text{Includes}(H, tst_1) = \text{true} \wedge \text{Includes}(H, tst_2) = \text{false} \wedge</math>  <math>\text{TST\_Before\_TST}(H, \text{Sous\_TST}))</math>                      else  <math>\emptyset</math> </p>

$$\begin{array}{l}
 \text{(III.162)} \\
 f_{\text{Union}}(tst_1, tst_2) := \left\{ \begin{array}{l}
 \begin{array}{l}
 \text{if } Equals(tst_1, tst_2) \vee Includes(tst_2, tst_1) \\
 \text{if } Includes(tst_1, tst_2) \\
 \text{if } ((tst_2.D.d_1.l_s = tst_1.A.a_n.l) \wedge (tst_2.D.d_1.t_s = \\
 tst_1.A.a_n.t_e + 1)) \vee ((tst_1.D.d_1.l_s = tst_2.A.a_n.l) \\
 \wedge (tst_1.D.d_1.t_s = tst_2.A.a_n.t_e + 1)) \\
 \\
 (tst_2.A \cup At\_Activity\_Activity(tst_1, \\
 Next\_Activity(tst_1, a_k), tst_1.A.a_{n1}).A, \\
 tst_2.D \cup At\_Activity\_Activity(tst_1, \\
 Next\_Activity(tst_1, a_k), tst_1.A.a_{n1}).D) \\
 \\
 \text{if } \exists (a_k(1 \leq k \leq n1) \in tst_1.A \wedge a_j(1 \leq j \leq n2) \in \\
 tst_2.A : Equals(At\_Activity\_Activity(tst_1, \\
 tst_1.A.a_1, a_k), At\_Activity\_Activity(tst_2, a_j, \\
 tst_2.A.a_{n2})) \\
 \\
 (tst_1.A \cup At\_Activity(tst_2, \\
 Next\_Activity(tst_2, a_k), tst_2.A.a_{n2}).A, \\
 tst_1.D \cup At\_Activity(tst_2, \\
 Next\_Activity(tst_2, a_k), tst_2.A.a_{n2}).D) \\
 \\
 \text{if } \exists (a_k(1 \leq k \leq n2) \in tst_2.A \wedge a_j(1 \leq j \leq n1) \in \\
 tst_1.A : Equals(At\_Activity\_Activity(tst_2, \\
 tst_2.A.a_1, a_k), At\_Activity\_Activity(tst_1, a_j, \\
 tst_1.A.a_{n1})) \\
 \text{else} \\
 \emptyset
 \end{array}
 \end{array} \right.
 \end{array}$$

(III.167)	$f_{Activities\_Difference}(tst_1, tst_2) := \{a \in D_a \mid a \in tst_1.A \wedge a \notin tst_2.A\}$	(IV.171)
(III.168)	$f_{Trips\_Difference}(tst_1, tst_2) := \{d \in D_d \mid d \in tst_1.D \wedge d \notin tst_2.D\}$	(IV.172)

Tableau IV.26 – Sémantique des opérations ensemblistes

Signature	Fonction d'interprétation
(III.169)	$f_{TST\_SpatTempNeighbour\_TST}(tst_1, tst_2, s, t) := existe\_tst(tst1, t) \wedge existe\_tst(tst2, t) \wedge ((\exists x, y \in A : x =$ $Activity\_At\_Time(tst_1, t) \wedge y = Activity\_At\_Time(tst_2, t) \wedge Neighbour(x.l, y.l, s)) \vee (\exists x \in A, y \in D :$ $x = Activity\_At\_Time(tst_1, t) \wedge y = Trip\_At\_Time(tst_2, t) \wedge Neighbour(x.l, y.path, s)) \vee (\exists x \in D, y \in A :$ $x = Trip\_At\_Time(tst_1, t) \wedge y = Activity\_At\_Time(tst_2, t) \wedge Neighbour(x.path, y.l, s)) \vee (\exists x, y \in D :$ $x = Trip\_At\_Time(tst_1, t) \wedge y = Trip\_At\_Time(tst_2, t) \wedge Neighbour(x.path, y.path, s))))$ (IV.173)
(III.170)	$f_{TST\_SpatTempNeighbor\_Point}(tst, p, s, t) := existe\_tst(tst, t) \wedge ((\exists a \in A : a = Activity\_At\_Time(tst, t)$ $\wedge Neighbour(a.l, p, s)) \vee (\exists d \in D : d = Trip\_At\_Time(tst, t) \wedge Neighbour(d.path, p, s)))$ (IV.174)
(III.171)	$f_{TST\_SpatTempNeighbor\_Polygone}(tst, l, s, t) := existe\_tst(tst, t) \wedge ((\exists a \in A : a = Activity\_At\_Time(tst, t) \wedge$ $Neighbour(a.l, l, s)) \vee (\exists d \in D : d = Trip\_At\_Time(tst, t) \wedge Neighbour(d.path, l, s)))$ (IV.175)
(III.172)	$f_{TST\_SpatTempNeighbour\_Region}(tst, r, s, t) := existe\_tst(tst, t) \wedge ((\exists a \in A : a = Activity\_At\_Time(tst, t) \wedge$ $Neighbour(a.l, r, s)) \vee (\exists d \in D : d = Trip\_At\_Time(tst, t) \wedge Neighbour(d.path, r, s)))$ (IV.176)

Tableau IV.27 – Sémantique des opérations de voisinage spatio-temporel

## 4 Conclusion

Dans ce chapitre, un modèle sémantique exprime formellement le comportement de la signature du type TST proposée dans le chapitre précédent. Ce modèle constitue une algèbre qui décrit le type TST. Pour cela, nous avons défini le domaine des valeurs du type TST et la sémantique de ses opérations. Cette sémantique est décrite formellement par le biais de fonctions d'interprétation associées à chaque opération. Grâce à cette méthode, nous proposons une structure formelle et des mécanismes pour modéliser rigoureusement les dynamiques spatio-temporelles et les sémantiques qui leur sont liées.

---

---

# Chapitre V

---

## Mise en oeuvre et expérimentation du TST

### 1 Introduction

Dans ce chapitre et dans un premier temps, nous proposerons l'architecture qui nous a servie pour la mise en oeuvre de notre type TST. Nous donnerons, aussi, une description de l'implémentation de notre type TST et des ses opérateurs sous le SGBD spatial PostgreSQL.

Dans un second temps, et pour valider notre travail, nous nous intéresserons à l'utilisabilité des fonctionnalités de notre type TST dans un cas pratique. Pour cela, nous décrirons comment le potentiel et les possibilités de notre modèle de représentation et de manipulation de trajectoires ont été expérimentés dans un contexte urbain, plus précisément dans le cadre d'une application d'une étude de la problématique de la mobilité au centre ville. Pour cela, nous présenterons le processus qui a servi à la génération du jeu de données utilisées pour l'expérimentation.

Finalement, nous présenterons la phase d'expérimentation au sein de laquelle nous avons formulé et appliqué un lot de requêtes, à différentes dimensions.

### 2 Mise en oeuvre

#### 2.1 L'environnement PostgreSQL/PostGIS

La mise en oeuvre de notre modèle nécessite le choix d'un SGBD disposant d'une extension spatiale, dans lequel nous implémentons notre nouveau type TST et le langage d'interrogation qui lui est associé. Actuellement, il existe sur le marché des acteurs ayant fournis un support spatial (extension spatiale sous forme de module ou cartouche de données) permettant la représentation, la gestion et l'analyse des données spatiales. Nous citons, à titre d'exemples, Oracle et son extension Oracle Spatial et MySQL et son extension spatiale MySQL Spatial et

PostgreSQL et son extension spatiale PostGIS.

Notre choix s'est porté sur le SGBD libre et open source PostgreSQL[22] et son extension spatiale PostGIS. Son caractère open source et son respect des spécifications de la norme OGC [64] ont appuyé ce choix. De plus, PostgreSQL offre des mécanismes qui permettent l'extension des fonctionnalités de son serveur par l'ajout de nouveaux types de données [22], comme les types composites ou les types de base, de fonctions utilisateurs, etc.

En effet, PostgreSQL est considéré comme un outil intéressant pour le prototypage rapide de nouvelles applications et de structures de stockage dans la mesure où son langage de requête SQL peut être étendu de plusieurs manières. En effet, il est possible de définir de nouveaux types de données et les opérations qui leur sont associés sous PostgreSQL, :

- en modifiant le code source et en créant ainsi de nouveaux types de base et des fonctions écrites en langage C ;
- en incorporant du code utilisateur par chargement dynamique. Le nouveau type ou la nouvelle fonction peut ainsi être chargé à la demande ;
- ou en créant, à l'aide de SQL et/ou d'un langage procédural comme PL/pgSQL, des types composites et leurs fonctions. Le nouveau type, sera tout simplement intégré dans la base où il a été défini. Cette dernière méthode, quoique moins rigoureuse, permet une expérimentation et un prototypage, facile et rapide, des nouveaux types créés.

### 2.2 Architecture proposée

Comme nous l'avons mentionné dans la section précédente, l'extension de PostgreSQL par un nouveau type de données peut s'effectuer par plusieurs méthodes. Dans l'état actuel de l'implémentation de notre prototype, nous avons codé le type TST comme un type composite (voir Figure V.1). Autrement dit, le nouveau type fera partie du système de types de PostgreSQL/PostGIS non pas en tant que type de base, mais en tant qu'un enregistrement représenté par l'ensemble de ces attributs. En dotant cet enregistrement par l'ensemble des opérations sur le type TST et en effectuant les manipulations sur le TST exclusivement par le biais de ces opérations, le type composite TST, ainsi réalisé, pourra être considéré comme type abstrait de données.

Les opérations associées aux TST sont implémentées comme des fonctions écrites en langage procédural PL/pgSQL qui est l'équivalent de pl/SQL dans PostgreSQL. Le code source du type TST utilise les types et les opérations de bases intégrées dans PostgreSQL, dont les types spatiaux (tel que le type *Linestring* ou *Point*) et temporels (tel que le type *Timestamp*). Nous utiliserons, le SIG libre, Quantum GIS, pour des fins de visualisations spatiales telle que la

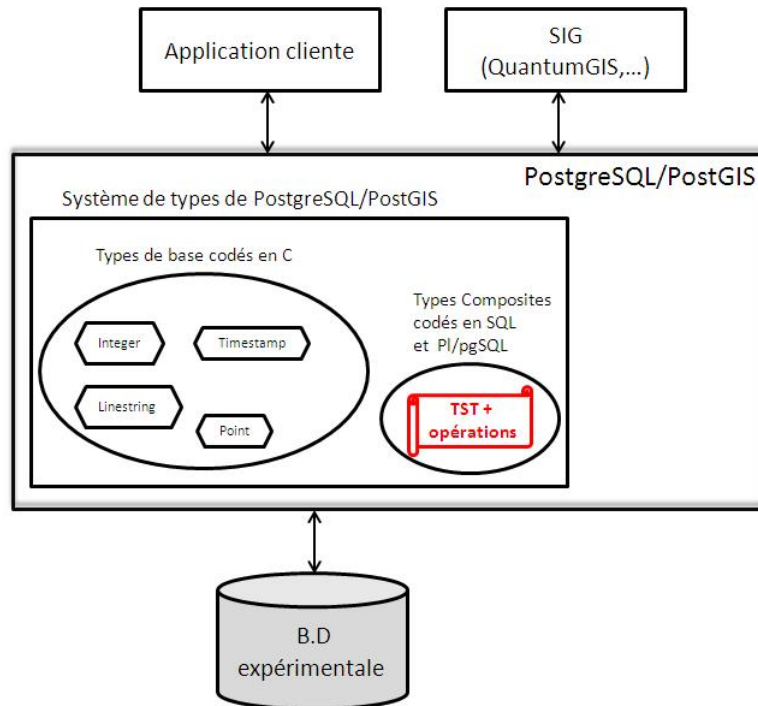


Figure V.1 – Architecture proposée

visualisation des itinéraires des trajectoires spatio-temporelles dans un espace spatial 2D, en cas de besoin.

## 2.3 Intégration du type TST dans PostgreSQL/PostGIS

### 2.3.1 Intégration des données

Nous rappelons que dans notre modèle, une valeur de type TST est composée d'activités et de déplacements. Nous avons, donc tout d'abord, créé les objets *activity* et *déplacement* comme des types composites. Ceci nous a permis de représenter ces notions comme des enregistrements composés d'une liste de champs, appartenant chacun à un type de base de PostgreSQL. Le type utilisé pour les attributs spatiaux est le type *geometry* et celui utilisé pour les attributs temporels est *timestamp*, ce dernier étant le plus approprié à la nature de l'application que nous allons utiliser pour notre expérimentation. Pour l'implémentation des activités et des déplacements, nous avons utilisé, comme suit, l'instruction "CREATE TYPE AS" :

```
CREATE TYPE activity AS
(
  ts timestamp,
```



```
te timestamp,  
l geometry,  
"type" character varying  
);
```

```
CREATE TYPE déplacement AS  
(  
ts timestamp,  
te timestamp,  
ls geometry,  
"mode" character varying[],  
path geometry);  
}
```

Une valeur de type TST est dans notre modèle, précisément, composée de deux ensembles ordonnés de même cardinalités, l'un est relatif aux activités et l'autre est relatif aux déplacements. Afin de garantir cette relation d'ordre, une valeur de type TST est représentée par deux tableaux dynamiques : l'un ayant des objets *activity* comme éléments et l'autre ayant des objets *déplacement* comme éléments (voir figure V.2). La manipulation de ces tableaux par les utilisateurs, qui ne doit se faire en principe qu'à l'aide des opérations fait en sorte que leur dimensions restent toujours les mêmes.

Pour l'implémentation du type TST, nous avons aussi utilisé, comme suit, l'instruction "CREATE TYPE AS", ce qui dresse le type TST comme un enregistrement mais auquel on offre un ensemble d'opérateurs de manipulations :

```
CREATE TYPE tst AS  
(a activity[],  
d déplacement[]);  
ALTER TYPE tst OWNER TO postgres;
```

L'utilisation de la structure de données tableau nous permettra une manipulation aisée et efficace des ensembles ordonnés au sein du TST. La comparaison entre différentes valeurs de TST sera aussi facile à effectuer à l'aide de cette structure de données. La structure tableau permet aussi de garantir la relations sémantique entre les événements du TST, vu que que les activités et les déplacements liés sémantiquement (e.g l'activité est l'objectif du déplacement) se trouvent à la même position dans le tableau.

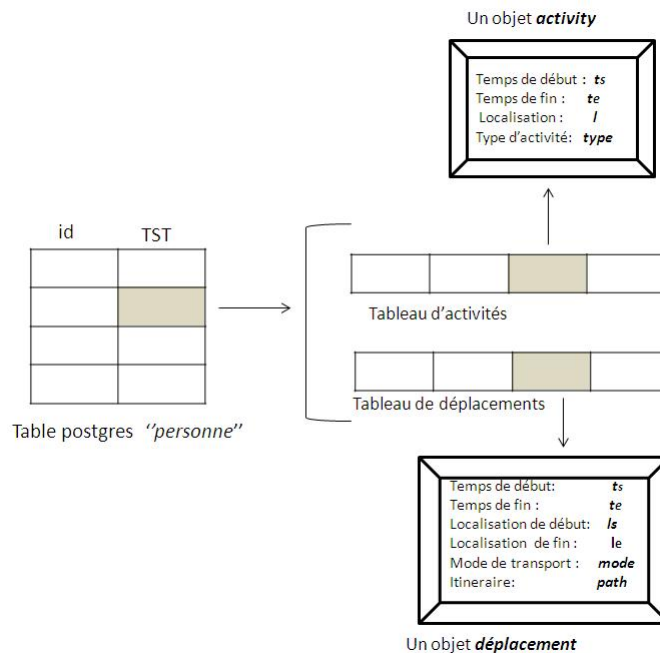


Figure V.2 – Structure de données d'un objet de type TST

### 2.3.2 Intégration des opérations

Pour chaque opération de manipulation du type TST proposée dans le chapitre précédent, un algorithme a été créé afin de l'opérationnaliser. Un ensemble de ces algorithmes, ceux qui nous ont semblé les plus intéressants à présenter, seront détaillés dans l'annexe A. Chacun des algorithmes proposés a été tourné par un code PL/pgSQL sous forme d'une fonction procédurale créée à l'aide de l'instruction "CREATE FUNCTION".

Nous donnons dans ce qui suit à titre illustratif, un exemple de codes relatifs à certaines opérations proposées.

En effet, afin de pouvoir vérifier la validité et l'intégrité des notions d'activités et de déplacements, nous avons implémenté en PL/pgSQL l'ensemble des opérations (*check\_activity* et *check\_trip*) qui vérifient si une activité ou un déplacement est valide. Le corps de ces fonctions sont écrites comme suit :

```
CREATE OR REPLACE FUNCTION check_deplacement(t deplacement) RETURNS boolean AS
$BODY$
BEGIN
    return t.ts < t.te and geometrytype(t.path) = 'LINESTRING'::text';
END;
$BODY$
```

## Chapitre V. Mise en oeuvre et expérimentation du TST

---

```
LANGUAGE 'plpgsql' VOLATILE STRICT COST 100;

CREATE OR REPLACE FUNCTION check_activity(t activity) RETURNS boolean AS
$BODY$
BEGIN
    return t.ts < t.te and geometrytype(t.l) = 'POINT'::text;
END;
$BODY$
LANGUAGE 'plpgsql' VOLATILE STRICT COST 100;
```

Afin de garantir l'intégrité de la base de données, nous avons aussi proposé une opération, `check_tst`, de vérification de la validité du type TST. Pour cela, il nous a fallu fixer une granularité temporelle. L'unité de temps dans notre implémentation a été fixée à 1 seconde. Le code source relatif à cette fonction est le suivant :

```
CREATE OR REPLACE FUNCTION check_tst(t tst) RETURNS boolean AS
$BODY$
DECLARE
valide boolean;
count integer;
n integer;
BEGIN
valide := true;
IF(NOT empty(t)) THEN
loop
count:=count+1;
IF count > n THEN
EXIT;
END IF;
IF count <> n THEN
IF not check_activity(t.a[count]) or not check_activity(t.a[count]) or
not Equals(t.a[count].l, t.d[count].le) or
not Equals(t.a[count].l, t.d[count +1].ls)
or t.d[count+1].ts <> t.a[count].te + '00:00:01' or
t.d[count].te + '00:00:01'<> t.a[count].ts THEN
valide := false;
EXIT;
```

```

        END IF;
    ELSE
        IF not check_activity(t.a[count] ) or not check_activity(t.a[count])
            or not Equals(t.a[count].l, t.d[count].le) or
            not Equals(t.a[count].l, t.d[count +1].ls) THEN
            valide := false;
            EXIT;
        END IF;
    END IF ;
end loop;
END IF ;
RETURN t;
END;
$BODY$
LANGUAGE 'plpgsql' VOLATILE STRICT;

```

Ces fonctions sont utilisées au sein d'autres fonctions à chaque fois qu'une activité, un déplacement ou une valeur de type TST est donné en argument et qu'il y a besoin de vérifier sa validité. A titre d'exemple, lors de l'alimentation d'un TST par un couple d'activité/déplacement, l'opération *ADD*, dont le code source est présenté ci-après, appelle les opérations de vérifications de validité mentionnées précédemment avant d'ajouter ce couple au TST.

```

CREATE OR REPLACE FUNCTION Add(t tst,a activity,d deplacement) RETURNS  tst AS
$BODY$
BEGIN
if (not check_activity(a) ) then
    RAISE EXCEPTION 'activité non valide';
end if;
if (not check_deplacement(d) ) then
    RAISE EXCEPTION 'déplacement non valide';
end if;
if (not check_tst(tst) ) then
    RAISE EXCEPTION 'TST non valide';
end if;
if((d.le = a.l and d.te + '00:00:01'= a.ts ) and ( length(t)=0 or
    (d.ts = t.a[length(t)].te + '00:00:01' and d.ls = t.a[length(t)].l) ) ) then

    t.a:=t.a|| a::activity;

```

```
t.d:=t.d|| d::deplacement;
RETURN t;
else
  if( not Equals(d.le, a.l)) then
    RAISE EXCEPTION 'erreur conditions sur couple d/a ';
  else
    if( d.te + '00:00:01' <> a.ts) then
      RAISE EXCEPTION 'erreur conditions sur couple d/a';
    else
      if((d.ts <> t.a[length(t)].te + '00:00:01')) then
        RAISE EXCEPTION 'erreur conditions sur couple d/a';
      else
        if(not equals(d.ls,t.a[length(t)].l)) then
          RAISE EXCEPTION 'erreur conditions sur couple d/a' ;
        end if;
      end if;
    end if;
  end if;
end if;

RETURN t;
END;
$BODY$
LANGUAGE 'plpgsql' VOLATILE STRICT;
```

### 3 Expérimentation

#### 3.1 Cadre expérimental

Nous présentons, dans cette section, le cas d'étude avec lequel nous avons évalué notre langage. Nous exposons, par la suite, le jeu de données utilisé lors de la validation de notre modèle ainsi que les différentes étapes du processus d'épuration des données initiales afin d'obtenir les données expérimentales finales.

### 3.1.1 Cas d'étude

Dans un contexte urbain, de nos jours, la mobilité au centre ville (C.V) présente des problématiques diverses dont les implications sont importantes à plusieurs dimensions. Le premier constat à relever concerne le double processus d'extension spatiale des villes, notamment par l'habitat périphérique, et la tertiarisation progressive des centres ville, ce qui donne lieu à la généralisation des déplacements radiaux périphérie-centre [20]. De plus, l'augmentation de la motorisation des ménages contribue à l'intensification des déplacements vers, à partir et dans les centres ville. Ceci implique des conséquences perceptibles à plusieurs niveaux.

En effet, à un niveau urbain, cette situation génère des difficultés de la circulation véhiculaire au niveau du centre ville mais également au niveau des routes radiales qui assurent l'accès. Les conséquences touchent également les difficultés liées aux stationnements à l'intérieur des centres ville, avec une inadéquation entre offre et demande et pertes de temps à la recherche de places disponibles, ce qui n'est pas sans avoir des effets sur la circulation. A un niveau énergétique et environnemental, cette augmentation des déplacements motorisés provoque une élévation de la consommation d'énergie et des nuisances environnementales.

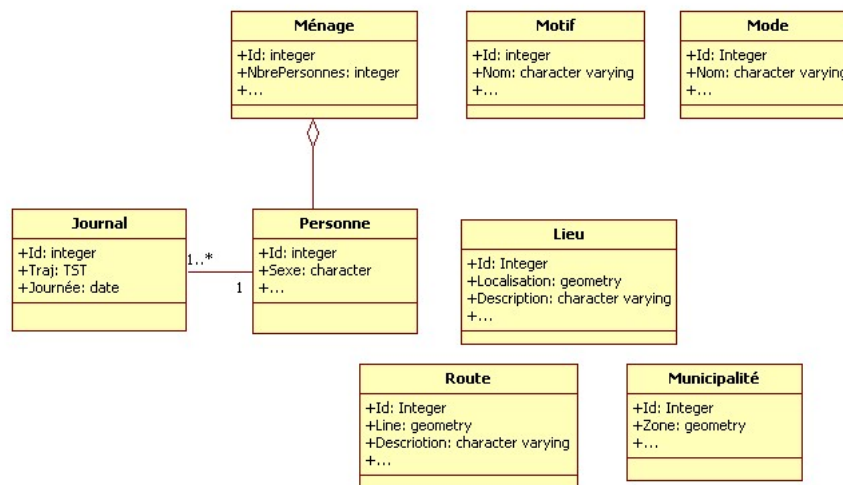
L'étude de la mobilité dans le centre ville permet aux décideurs et aux planificateurs d'être mieux outillés pour dégager et analyser ces problématiques. Ces connaissances seront à même d'aider les décideurs à mettre en place des politiques adéquates en faveur d'une planification urbaine et des déplacements susceptibles d'améliorer le fonctionnement des centres ville.

Pour ce faire, l'étude du comportement des déplacements, vers, depuis et au sein du centre ville peut apporter des connaissances intéressantes à ce sujet. L'observation et l'analyse de ces données peut porter sur différents aspects dont notamment :

- le profil des usagers du centre ville, qu'ils soient résidents ou visiteurs,
- les pratiques d'utilisations des différentes catégories de routes (ponts, routes, etc.) pour accéder ou sortir du centre ville,
- l'origine, la destination, la durée et la longueur de leurs trajets,
- le flux d'automobilistes du et vers le centre ville pendant les heures de pointes,
- les motifs des déplacements et la nature des activités exercées au centre ville,
- les pratiques de stationnement pour les déplacements en voiture particulière,
- les possibilités de co-voiturage, etc.

Procurer la possibilité de modéliser aisément la configuration spatio-temporelle et les activités d'un tel échantillon d'utilisateur, et d'émettre des requêtes à la fois riches et faciles à formuler, serait, ainsi, d'un apport majeure pour les décideurs.

Dans la section suivante, nous présentons un exemple d'application qui, à l'aide de notre



**Figure V.3** – Squelette proposé pour un modèle de la base de données expérimentale

type TST, modélise le comportement de mobilité de différents individus en se basant sur leurs journaux d'activités pour une journée particulière.

### 3.1.2 Modèle de base de données pour l'étude de la problématique du centre ville

Afin d'évaluer notre type TST, nous proposons un modèle conceptuel (voir figure V.3) pour une application qui vise à analyser la mobilité d'un groupe de personnes et à étudier cette mobilité par rapport à la problématique du centre ville. Un exemple similaire à ce modèle a été proposé dans nos publications [16, 17]. Dans ce modèle, qui est un diagramme de classe, nous distinguons trois catégories de classes.

La première catégorie, regroupe les classes *Route*, *Lieu* et *Municipalité*. Ce sont les classes qui modélisent l'environnement dans lequel les personnes se déplacent. La classe *Municipalité* décrit un ensemble de régions de type *Polygon*. Parmi ces régions, on peut retrouver la région associée au centre ville. La classe *Route* représente l'ensemble de géométries de type *Linestring* le long desquelles les personnes sont censées circuler. Ce sont, en fait, les segments de route du réseau routier étudié. L'attribut *Description* de la classe *Route* donne une description textuelle de la nature de chaque segment de route qui peut être, à titre d'exemples, une autoroute ou un tunnel. Enfin, la classe *Lieu* décrit différentes localisations spatiales de différents endroits connus et ayant un intérêt pour l'application. À titre d'exemple, on peut retrouver les positions de centres de loisirs situés au centre ville, d'hôpitaux ou de stations de métro. L'attribut *Description* de cette dernière classe décrit textuellement la nature de ces lieux.

La seconde catégorie regroupe les classes *Motif* et *Mode*. Ce sont des classes qui représentent

des informations sur les raisons pour lesquelles les déplacements ont eu lieu, par exemple pour faire du "shopping" ou pour aller travailler, et sur les modes de déplacements que les personnes peuvent utiliser pour se déplacer comme la marche à pied, la voiture ou le transport public.

Finalement, la dernière catégorie de classe est composée des classes *Ménage*, *Personne* et *Journal*. La classe *Ménage* décrit les caractéristiques d'un ménage. Les attributs de cette classe peuvent être le nombre de personnes qui composent ce ménage, le nombre de véhicules en leur possession, etc. Un ménage est composé de personnes. La classe *Ménage* est donc modélisée comme un agrégat de personnes. La classe *Personne* décrit les informations qui peuvent caractériser une personne comme son sexe, son âge, etc. A chaque personne est associée un ou plusieurs journaux d'activités correspondant à une journée particulière. Un journal possède un identifiant et est relatif à une journée particulière de type *date*. Le journal d'activités d'une personne pendant une journée est décrit par l'attribut *Traj* qui est une trajectoire spatio-temporelle sémantiquement riche de type *TST*. Ainsi, dans ce modèle, nous avons pu utiliser notre type *TST* pour modéliser d'une façon aisée et unifiée le comportement de déplacements et d'activités journaliers d'un groupe de personnes.

Le modèle proposé doit être alimenté par des données issues de données benchmark et ceci pour pouvoir construire une base de données expérimentale sur laquelle nous pouvons émettre nos requêtes. Dans la section qui suit, nous décrivons les étapes de création d'une telle base de données.

## 3.2 Jeu de données

### 3.2.1 Données benchmark

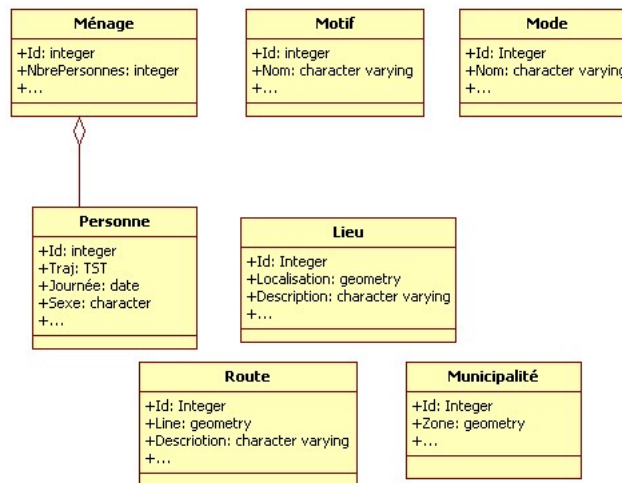
Lors de notre expérimentation, nous avons utilisé des données benchmark. Nous allons décrire dans ce qui suit le contenu de ces données ainsi que les étapes suivies lors du processus d'épuration de ces données.

Les données initiales que nous avons utilisé pour la création de notre base de données expérimentale sont les suivantes :

1. des données provenant d'une banque de données issue d'une enquête origine-destination (O-D) réalisée dans la région du Québec en 2001,
2. des données illustrant les zones relatives aux municipalités de la région de Québec,
3. et un réseau routier de la région de Québec.

Voici les détails quant au contenu de ces données.



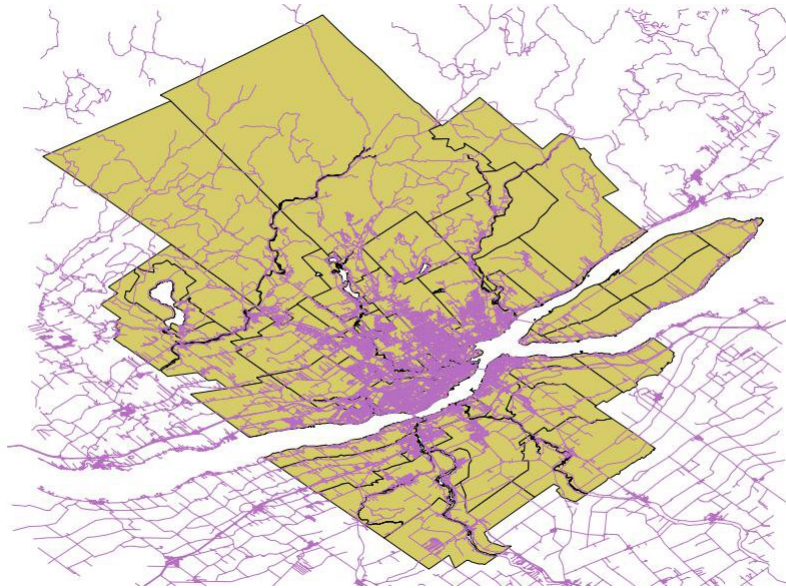


**Figure V.4** – Modèle final adopté de la base de données expérimentale

**Enquête origine-destination** Les enquêtes O-D permettent de tracer le portrait d’une population et des ses déplacements dans une région donnée. Ces enquêtes sont habituellement réalisées auprès d’un échantillon de ménages de la région visée. Elles contiennent une description des déplacements et activités de chacun des membres des ménages au cours d’une journée. Les données sont par la suite codifiées, traitées, puis utilisées pour différentes études.

La base de données que nous avons utilisée est, en fait, une base de déplacements de personnes qui ont répondu à une enquête O-D conduite en 2001 par la Société de Transport de la Communauté Urbaine de Québec (STCUQ) [19]. Chaque ligne de cette base contient des informations caractérisant le déplacement d’une personne. Ces données sont décrites dans l’annexe B. En effet, nous pouvons remarquer que dans le cas des données qui nous ont été fournies, l’enquête O-D concerne des personnes différentes chacune questionnée lors d’une unique journée. Une personne ne figure, donc, qu’une unique fois dans la base benchmark O-D et ne possèdera donc qu’une seule trajectoire réalisée à cette date. Nous pouvons, donc, dans notre cas, simplifier le modèle de la figure V.3 en éliminant la classe *Journal* qui devient inutile et en intégrant les informations *Traj* et *Journée* comme attributs dans la classe *Personne* (voir figure V.4).

**Zone d’étude** Vu que notre cas d’étude porte sur la problématique du C.V, nous avons eu besoin de données géométriques de type *Polygon* qui décrivent la zone d’étude. Une base décrivant, la géométrie des différentes municipalités des communes urbaines de la région de Québec est utilisée pour ce but (voir figure V.5). La municipalité de Charlesbourg, représentera la zone géométrique du C.V et les requêtes sur la base de données expérimentale porteront



**Figure V.5** – Zone d'étude : la communauté urbaine du Québec et son réseau routier

principalement sur cette zone.

**Le réseau routier** Le réseau routier de la région urbaine de Québec (voir figure V.5) nous servira dans notre expérimentation pour la visualisation des résultats des requêtes. Nous nous en servons aussi pour extraire des données spatiales qui alimentent les attributs de type TST de notre base expérimentale.

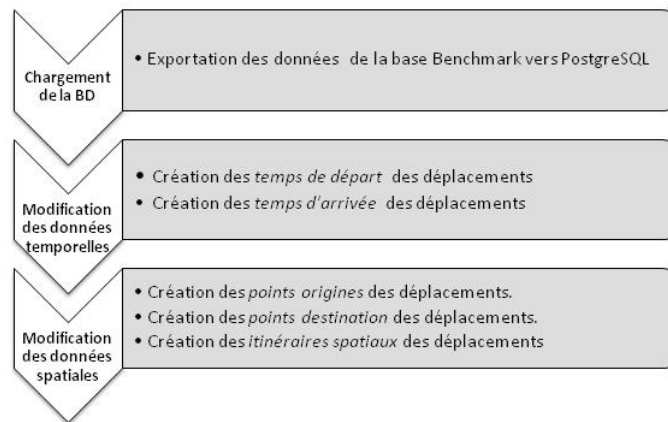
### 3.2.2 Processus d'épuration des données benchmark

Afin de pouvoir être exploitée selon notre modélisation, les données obtenues ont subies un travail d'épuration. Les phases de ce processus sont illustrée par la figure V.6. Plus de détails sur ce processus sont fournis dans l'annexe B.

A l'issue de ce travail de pré-traitement, nous avons obtenu une base de données expérimentale comportant sept tables (conformément au diagramme de classe de la figure V.4). La table ménage comporte 4144 familles. La table personne comporte un échantillon de 9095 enregistrements correspondant chacun à une personne.

## 3.3 Résultats expérimentaux

Cette section illustre le potentiel de notre langage à travers un ensemble de requêtes émises sur notre base de données expérimentale et ceci dans le contexte du cas d'étude choisi, à savoir



**Figure V.6** – Étapes du processus de pré-traitement des données Benchmark

la problématique de la mobilité au C.V. Cet ensemble de requêtes est, tout d'abord, formulé textuellement puis implémenté avec le langage PL/pgSQL [22] qui est le langage de requête de l'environnement PostgreSQL. Le résultat obtenu est présenté sous forme tabulaire et visualisé à l'aide du SIG QuantumGIS quand ceci est possible.

**Requête 1** "Combien de personnes passent par le C.V entre deux séances de travail?"

Pour répondre à cette requête, il nous faut tester sur les personnes travaillant hors du C.V. Pour ce faire, nous pouvons tester sur l'attribut "lieuactivprincapl" (voir tableau B.2 de l'annexe B) qui indique le lieu d'activité principale de chaque personne et vérifier qu'il n'est pas contenu au C.V. Si cette information ne figure pas dans la base, nous pouvons à l'aide de nos opérations sur le TST trouver le lieu de l'activité de type "travail" et tester ainsi sur ce lieu. Afin d'avoir l'itinéraire entre deux activités de type "travail", nous utilisons l'opération de restriction sémantique *At\_Activity\_Activity*. Cette opération nous permet de récupérer un bout de la trajectoire spatio-temporelle de type TST des personnes en questions. Ce bout de trajectoire doit être délimité par la première activité de type travail (récupérée par l'opération sémantique *Activity\_With\_Type*) et par l'activité de type travail qui suit directement cette activité (récupérée par l'opération *Next\_Activity\_With\_Type*). L'opération de topologie spatiale *TST\_Cross\_Region* nous permet ensuite de tester le croisement entre cette sous-trajectoire et le C.V. Ainsi, dans cette formulation, nous avons montré que l'utilisation combinée, simple et efficace des différents opérateurs que nous offrons et ceci à des niveaux aussi bien spatiaux que sémantiques, nous a permis de répondre à cette requête.

```
SELECT count(*)
FROM personne as pr, municipalites AS cv
```

```

WHERE cv.nom_mun = 'Charlesbourg'
AND Activities_With_Type_Count(pr.traj,'1') >=2
AND not contains(cv.zone,pr.lieuactiprincipal)
AND TST_Cross_Region(
    At_Activity_Activity(pr.traj, Activity_With_Type(pr.traj,'1')),
    Next_Activity_With_Type(pr.traj,'1', Activity_With_Type(pr.traj,'1')))
, cv.zone);

```

La figure V.7 montre, sous l'outil QGIS [65], les itinéraires des personnes vérifiant cette requête ainsi que les points représentant les deux activités avant et après lesquelles les personnes sont passées par le C.V. Ces informations de types géométriques peuvent être récupérées en ajoutant à la partie SELECT de la formulation précédente, en plus du nombre de personnes, les itinéraires des sous-trajectoires récupérées ainsi que le premier et le dernier point de cette sous-trajectoire. On obtient ainsi la partie SELECT formulée comme suit :

```

SELECT Itinerary(At_Activity_Activity(pr.traj, Activity_With_Type(pr.traj,'1')),
    Next_Activity_With_Type(pr.traj,'1',Activity_With_Type(pr.traj,'1'))),

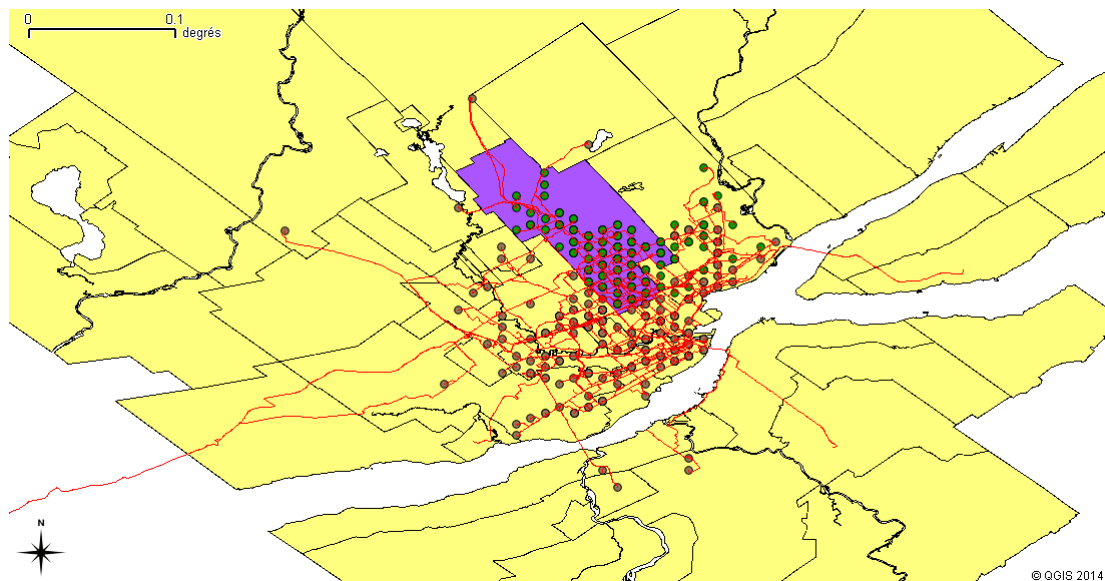
    First_Point(At_Activity_Activity(pr.traj, Activity_With_Type(pr.traj,'1')),
    Next_Activity_With_Type(pr.traj,'1',Activity_With_Type(pr.traj,'1'))),

    Last_Point(At_Activity_Activity(pr.traj, Activity_With_Type(pr.traj,'1')),
    Next_Activity_With_Type(pr.traj,'1',Activity_With_Type(pr.traj,'1')))

```

Dans notre proposition de solution à cette requête, nous n'avons testé que sur l'itinéraire séparant les deux premières activités de type "travail" d'une personne. Néanmoins, nous pouvons facilement, reprendre cette formulation et la généraliser au cas où les trajectoires testées sont telles que plus de deux activités de type "travail" ont lieu. Ceci se concrétiserait par un code PL/pgSQL, dans lequel, il y aura parcours de la trajectoire, selon un critère sémantique, et ceci à l'aide de l'opération *Next\_Activity\_With\_Type*. La formulation deviendrait, un peu plus longue bien sûr mais elle reste à notre avis facilement réalisable.

**Requête 2** "Quelles sont les personnes qui travaillent et font du magasinage au C.V? font-elles leurs achats plutôt avant ou après le travail et à quelle fréquence?"



**Figure V.7** – Visualisation des résultats de la requête 1

Cette requête pourrait être posée afin d'étudier les activités commerciales au sein du C.V en analysant le comportement des individus travaillant et faisant leurs achats au C.V et les relations entre les deux activités.

En utilisant la base de données benchmark de départ, qui ne prend pas en considération notre type TST, l'implémentation de cette requête est compliquée. Elle nécessite, en effet, la formulation de plusieurs requêtes imbriquées faisant le parcours des enregistrements de la base personne par personne. Pour chaque personne, il faudrait chercher le premier déplacement dont le motif est le "travail" (codé par le nombre 1). En cas de succès, il faudrait reparcourir la table personne pour compter les nombres de déplacements ayant le motif "magasinage" (codé par le nombre 5) qui précèdent et ceux qui suivent temporellement le déplacement de motif "travail".

En utilisant notre type TST et notre langage de manipulation, la requête pourrait être formulée comme suit :

```
SELECT pr.id,  
       Activities_With_Type_Count( At_Activity_Activity(pr.traj,  
                                                       First_Activity(pr.traj),  
                                                       Activity_With_Type(pr.traj,'1')), '5'),  
  
       Activities_With_Type_Count( At_Activity_Activity(pr.traj,  
                                                       Activity_With_Type(pr.traj,'1'),  
                                                       Last_Activity(pr.traj)), '5')
```

```

FROM personne AS pr, municipalites AS cv
WHERE length(pr.traj) >=2 --Il existe au moins une activité dans le TST
AND cv.nom_mun = 'Charlesbourg'
--Il existe des activités de type "shopping" dans le C.V
AND Activities_With_Type(pr.traj,'5') &&
                                Activities_At_Region(pr.traj, cv.zone)
--Il existe des activités de type "travail" dans le C.V
AND Activities_With_Type(pr.traj,'1') &&
                                Activities_At_Region(pr.traj, cv.zone) ;

```

En effet, nous avons obtenu une formulation aisée en combinant l'utilisation de différentes opérations sémantiques sur le type TST. En effet, nous avons restreint notre recherche dans la table *personne*, aux enregistrements ayant des activités de type "<travail>" et de type "magsinage" au sein du C.V. Ceci est obtenu en vérifiant l'intersection non vide (l'opération &&) des tableaux retournés par les opérateurs *Activities\_With\_Type* et *Activities\_At\_Region*. Les fréquences des activités "magsinage" avant le travail est obtenu en calculant le nombre d'activités de ce type dans la sous-trajectoire "traj\_avant". Cette sous-trajectoire, représente l'ensemble des activités et déplacements de la trajectoire qui précèdent la première activité de type "travail". Cette sous trajectoire est obtenue en appliquant notre opération de restriction sémantique *At\_Activity\_Activity*. Le même raisonnement est appliqué pour obtenir la sous-trajectoire <traj\_après>.

**Requête 3 :** Quelles sont les personnes qui terminent leur journée en quittant le C.V à partir de leur lieu de travail après 17h et en voiture?

Cette requête pourra donner des précisions sur le comportement de mobilité des personnes qui travaillent dans le C.V et qui le quittent en étant motorisé.

Pour répondre à cette requête, il faut vérifier les conditions suivantes :

1. le dernier déplacement d'une personne part d'un lieu, d'activité *p*, situé au C.V ;
2. à ce lieu, *p*, la personne a exercé l'activité "travail", cette dernière est l'avant dernière activité de la personne durant cette journée ;
3. le dernier déplacement se dirige vers un lieu situé hors du C.V ;
4. le dernier déplacement débute temporellement après 17h ;
5. le dernier déplacement est effectué avec le mode "voiture".

## Chapitre V. Mise en oeuvre et expérimentation du TST

---

Comme pour la requête précédente, utiliser la base de données benchmark pour formaliser cette requête nécessite un parcours par personne qui doit être réitéré pour vérifier le motif du déplacement qui précède le dernier déplacement d'une personne.

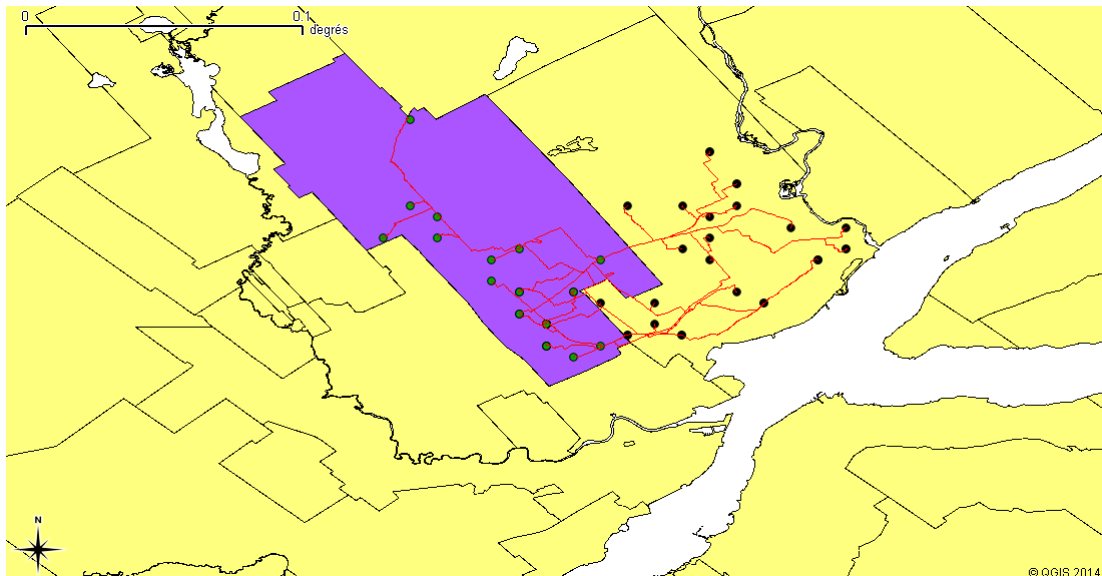
En utilisant notre base de données expérimentale et notre langage, nous proposons plusieurs solutions d'implémentation de cette requête.

### 1ère solution

Nous pouvons utiliser les opérations de restrictions sémantiques, plus particulièrement l'opération *At\_Activity\_Activity* pour restreindre la trajectoire de la personne au dernier couple activité/déplacement et l'avant dernier couple activité/déplacement. Ainsi, nous pouvons requêter sur ces sous-trajectoires de taille égale à 1 et tester si l'unique activité exercée est de type "travail" et si le dernier point de la sous-trajectoire est contenu dans le C.V.

```
SELECT pr.id
FROM (SELECT pr2.id AS pr2, At_Activity_Activity(pr2.traj,
                                                Nth_Activity( pr2.traj, length(pr2.traj)-1),
                                                Nth_Activity(pr2.traj, length(pr2.traj)-1)) AS AtAA
      FROM personne AS pr2) AS sous_requete,
personne AS pr, municipalites AS cv
WHERE
pr2 = pr.id
AND length(pr.traj) >=2 --Il existe au moins 2 activités
AND cv.nom_mun = 'Charlesbourg'
-- vérification des conditions 1 et 4
AND contains(cv.zone,Last_Point(AtAA))
AND Activities_With_Type_Count(AtAA , '1') = 1
--vérification de la condition 2 et 3
AND Last_Time(AtAA)::time >= '17:00:00'::time
AND not contains(cv.zone,Last_Point(pr.traj))
--vérification de la condition 5
AND Trips_With_Mode_Count(At_Activity_Activity(pr.traj, Last_Activity(pr.traj),
                                                Last_Activity(pr.traj)), '1') = 1
ORDER BY pr.id;
```

En ajoutant aux objets de la sélection les deniers points de chaque sous-trajectoire sélectionnée et l'itinéraire de la dernière sous-trajectoire du TST de chaque personne, nous pouvons visualiser (voir Figure V.8) les lieux des activités travail au C.V (mentionnés en vert) et les



**Figure V.8** – Visualisation des résultats de la requête 3

lieux de destination hors du C.V (mentionnés en noir) et l'itinéraire entre ces lieux qui a lieu en voiture.

### 2ème solution

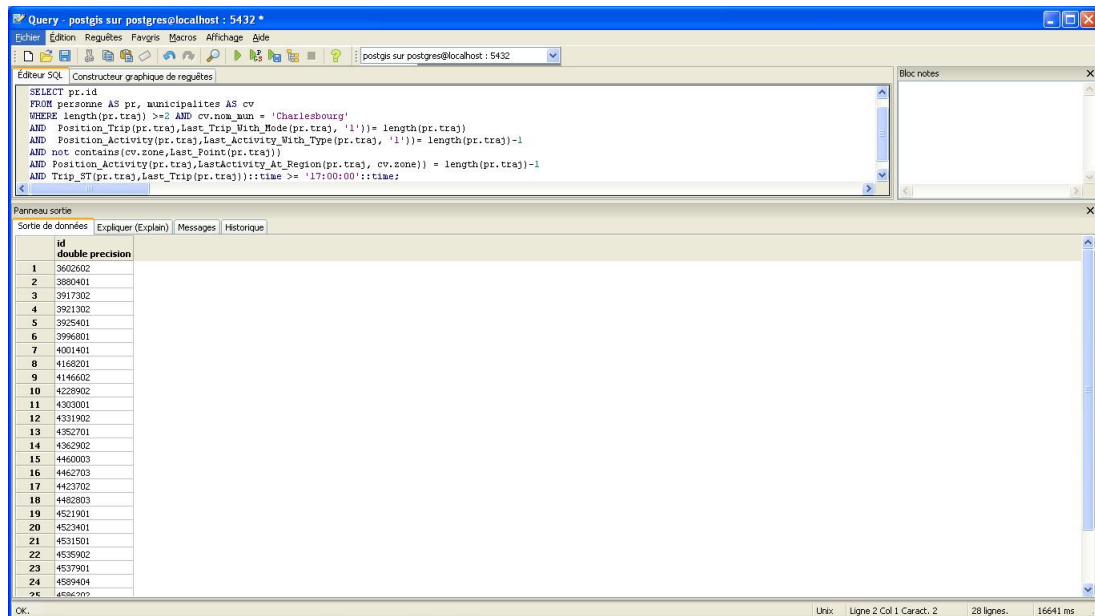
La deuxième solution proposée est d'utiliser les opérations de recherche sémantiques pour tester les caractéristiques des déplacements et activités en question. Ainsi, on pourra, par exemple, récupérer avec l'opération *Last\_Activity\_With\_Type* une activité et tester si elle est bien l'avant dernière activité dans le chaînage des activités du TST, c'est-à-dire de position égale à la taille de la trajectoire -1. De même on pourra récupérer un déplacement avec *Last\_Trip\_With\_Mode* et tester si ce déplacement est le dernier déplacement effectué dans le chaînage du TST.

```

SELECT pr.id
FROM personne AS pr, municipalites AS cv
WHERE length(pr.traj) >=2 AND cv.nom_mun = 'Charlesbourg'
  --vérification de la condition 5
AND Position_Trip(pr.traj, Last_Trip_With_Mode(pr.traj, '1')) = length(pr.traj)
  --vérification de la condition 2
AND Position_Activity(pr.traj, Last_Activity_With_Type(pr.traj, '1')) =
                                length(pr.traj)-1
  --vérification de la condition 3
AND not contains(cv.zone, Last_Point(pr.traj))
  
```



## Chapitre V. Mise en oeuvre et expérimentation du TST



```
SELECT pr.id
FROM personne AS pr, municipalites AS cv
WHERE length(pr.traj) >= 1 AND cv.nom_aum = 'Charlesbourg'
AND Position_Trip(pr.traj, Last_Trip-With_Mode(pr.traj, '1')) = length(pr.traj)
AND Position_Activity(pr.traj, Last_Activity-With_Type(pr.traj, '1')) = length(pr.traj)-1
AND NOT contains(cv.zone, Last_Point(pr.traj))
AND Position_Activity(pr.traj, LastActivity_At_Region(pr.traj, cv.zone)) = length(pr.traj)-1
AND Trip_ST(pr.traj, Last_Trip(pr.traj))::time >= '17:00:00'::time;
```

id	double precision
1	3602602
2	3880401
3	3917302
4	3921302
5	3925401
6	3996801
7	4001401
8	4168201
9	4146602
10	4228902
11	4303001
12	4331902
13	4352701
14	4362902
15	4460003
16	4463703
17	4423702
18	4482803
19	4521901
20	4523401
21	4531501
22	4535902
23	4537901
24	4559404
25	4562902

Figure V.9 – Résultats tabulaires de la requête 3

--vérification de la condition 1

```
AND Position_Activity(pr.traj, LastActivity_At_Region(pr.traj, cv.zone)) =
length(pr.traj)-1
```

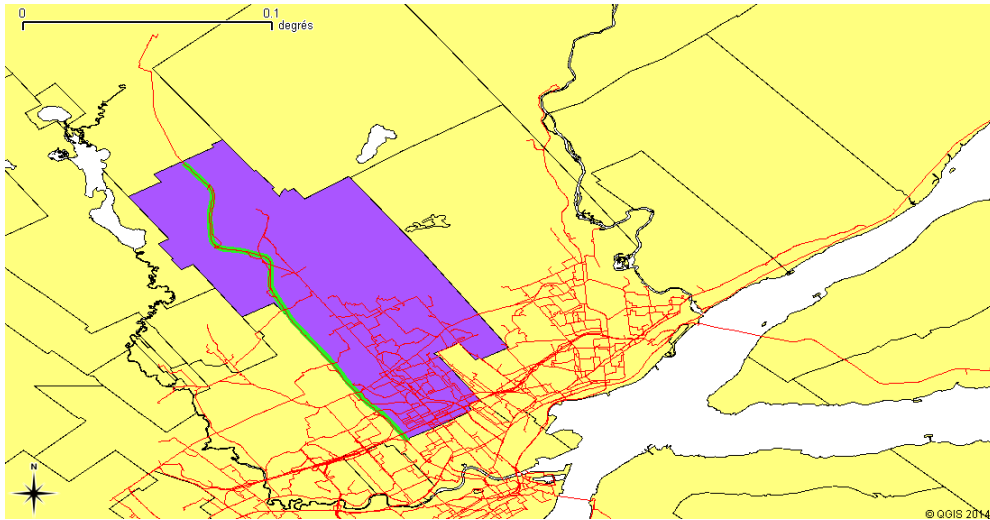
-- vérification de la condition 4

```
AND Trip_ST(pr.traj, Last_Trip(pr.traj))::time >= '17:00:00'::time;
```

Les deux solutions formulées donnent le même résultat, c'est-à-dire les mêmes personnes au nombre de 28 enregistrements (voir Figure V.9).

**Requête 4:** Quelles sont les personnes qui empruntent les autoroutes du C.V?

Cette requête est un exemple de l'utilité et de l'utilisation simple des opérations de topologies spatiales entre un objet TST et un objet géométrique. En effet, l'opération *TST\_Contains\_Polyline*, appliquée aux TST des différentes personnes de la base expérimentale et à des segments du réseau, permet de tester si l'itinéraire des personnes passe bien par ces segments de route. Il faut, par contre, sélectionner au préalable les segments de routes qui passent par le C.V et qui sont décrits comme étant des autoroutes. Ceci est effectué sous la forme d'une sous-requête identifiée par l'alias "Autoroute". La figure V.11 donne un aperçu sur les enregistrements correspondant aux personnes retournées comme résultat à cette requête. Nous donnons aussi un aperçu sur la visualisation des itinéraires de ces personnes (mentionnés en rouge) qui comme le montre la figure V.10 passent bien par les autoroutes (mentionnés en vert) au sein du C.V.



**Figure V.10** – Visualisation des résultats de la requête 4

```

SELECT pr.id , Itinerary(pr.traj)
FROM ( SELECT Line from Route, municipalites as cv
        WHERE descriptio='Autoroute' AND cv.nom_mun= 'Charlesbourg'
        AND (contains(cv.zone, Line) OR crosses(cv.zone,Line))
        ) AS Autoroute,
        personne AS pr, municipalites as cv
WHERE cv.nom_mun = 'Charlesbourg'
AND TST_Contains_Polyline(pr.traj,Autoroute.Line);

```

**Requête 5** : Quelle est la distance maximale, au C.V, qui sépare tous les itinéraires des personnes qui se sont déplacées intégralement dans le C.V entre 10 heures et minuit ?

Cette requête montre essentiellement le potentiel spatio-temporel de notre langage par l'utilisation combinée et aisée des opérations de restriction temporelle et des opérations spatiales telles que les opérations de topologie spatiale et les opérations de calcul spatial.

Pour cette requête et pour chaque personne, on s'intéresse à la partie de son itinéraire qui s'est déroulée intégralement au C.V entre dix heures et minuit. Pour ce faire, et en utilisant l'opération de restriction temporelle *At\_Interval*, nous restreignons le TST de chaque personne à cet intervalle-là. Il suffit par la suite de tester, à l'aide de l'opération *TST\_Inside\_Region* si cette sous-trajectoire est contenue dans le C.V. La distance maximale entre ces sous-trajectoires est le résultat de l'opération d'agrégation *Max* appliquée aux distances entre les sous-trajectoires. Les distances entre les trajectoires sont données par l'opérateur sur les *TST\_Distance\_TST*.

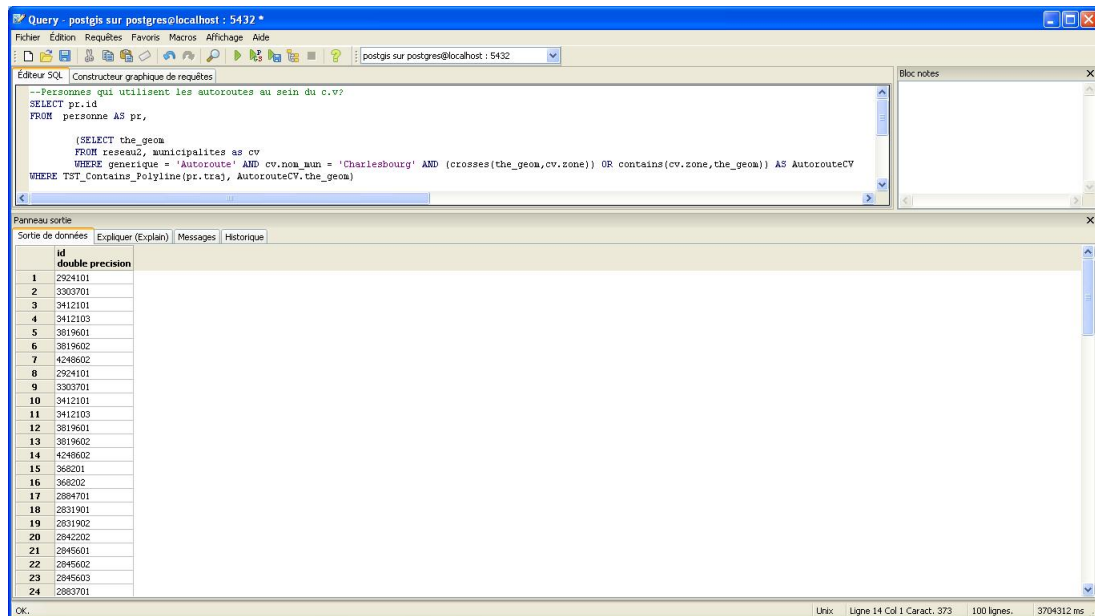


Figure V.11 – Résultats tabulaires de la requête 4

```
select distinct Max(TST_Distance_TST( At_Interval(pr1.traj, ('10:00', '00:00')),
                                     At_Interval(pr2.traj, ('10:00', '00:00'))))
from personne pr1 , personne pr2, municipalites AS cv
where TST_Inside_Region( At_Interval(pr1.traj, ('10:00', '00:00')), cv.zone)
AND   TST_Inside_Region( At_Interval(pr2.traj, ('10:00', '00:00')), cv.zone);
```

**Requête 6 :** Quelles sont, pour chaque personne qui se déplace vers le C.V au début de la journée, la durée et la distance de ce déplacement? et que représente cette distance par rapport aux distances totales de leur trajectoires?

Il s'agit, ici, de calculer la distance et la durée du premier déplacement de la trajectoire de chaque personne. Cette personne doit avoir commencé ce déplacement d'un endroit hors du C.V et l'avoir terminé à un endroit au sein du C.V. Pour ce faire, nous effectuons une opération de restriction sémantique de chaque trajectoire à la sous trajectoire, *sous\_traj*, composée du premier couple activité/déplacement. Cette sous-trajectoire doit vérifier les conditions précédemment décrites. Le lieu de départ et le lieu d'arrivée de la trajectoire *sous\_Traj* sont récupérés par les opérations *First\_Point* et *Last\_Point*. La distance parcourue est calculée en ayant recours à l'opération spatiale sur les TST *Spatial\_Length*. La durée du déplacement, quant à elle, est donnée par l'opération temporelle *Trips\_Duration* appliquée à la sous-trajectoire *sous-traj*.

```
SELECT Spatial_Length(sous_traj), (Trips_Duration(sous_traj)),
```

The screenshot shows a PostgreSQL query editor window titled "Query - postgres sur postgreslocalhost : 5432". The query is as follows:

```
SELECT Spatial_Length(sous_traj), (Trips_Duration[sous_traj]), (Spatial_Length(sous_traj)* 100)/Spatial_Length(traj)
FROM ( SELECT At_Trip_Trip(pr.traj), First_Trip(pr.traj), First_Trip(pr.traj)) AS sous_traj, pr.traj as traj
FROM personne AS pr, municipalites AS cv
WHERE cv.nom_mun = 'Charlesbourg'
AND not contains(cv.zone, First_Point( At_Trip_Trip(pr.traj, First_Trip(pr.traj), First_Trip(pr.traj))))
AND contains(cv.zone, Last_Point( At_Trip_Trip(pr.traj, First_Trip(pr.traj), First_Trip(pr.traj))))
) as sous_requete;
```

The results are displayed in a table with the following columns: `spatial_length double precision`, `trips_duration interval`, and `?column? double precision`. The table contains 25 rows of data.

	spatial_length double precision	trips_duration interval	?column? double precision
1	0.0731506861929564	00:08:48	100
2	0.154875819822499	00:14:18	99.99999999999999
3	0.137264067336177	00:13:54	98.7133466104695
4	0.170188071572952	00:20:00	100
5	0.181433637534316	00:15:24	100
6	0.118937374736293	00:13:24	100
7	0.11067482304273	00:11:42	100
8	0.060485530220922	00:08:00	100
9	0.0811255967974747	00:08:54	100
10	0.0937110794657408	00:12:30	95.1308442869069
11	0.0960984763252191	00:10:12	100
12	0.116909377482775	00:13:18	99.99999999999999
13	0.0727296644202252	00:09:18	100
14	0.260318611552659	00:24:06	87.4039680103597
15	0.156770703471956	00:14:48	88.0189843501474
16	0.136282190302348	00:33:00	100
17	0.173952191717108	00:17:42	95.1872715941506
18	0.089437541913027	00:09:18	100
19	0.161714478102566	00:16:30	99.99999999999999
20	0.053211618624806	00:03:00	99.99999999999999
21	0.0538336272672807	00:00:12	99.99999999999999
22	0.0710729789990595	00:04:54	100
23	0.12938046442293	00:13:24	93.6421358432711
24	0.0553214153381626	00:07:00	100
25	0.003432328341114	00:10:06	61.2880871587447

Figure V.12 – Résultats tabulaires de la requête 6

```
(Spatial_Length(sous_traj)* 100)/Spatial_Length(traj)
FROM (SELECT At_Trip_Trip(pr.traj, First_Trip(pr.traj), First_Trip(pr.traj))
      AS sous_traj, pr.traj as traj
FROM personne AS pr, municipalites AS cv
WHERE cv.nom_mun = 'Charlesbourg'
AND not contains(cv.zone, First_Point(At_Trip_Trip(pr.traj,First_Trip(pr.traj),
                                                    First_Trip(pr.traj))))
AND Contains(cv.zone, Last_Point(At_Trip_Trip(pr.traj,First_Trip(pr.traj),
                                                    First_Trip(pr.traj))))
) as sous_requete;
```

La figure V.12 illustre les résultats d'exécution de cette requête.

**Requête 7:** Quelles sont les possibilités de co-voiturage vers le C.V au début de la journée?

La réponse à cette requête peut prendre la forme d'un couple de personnes. Ces personnes doivent vérifier les conditions suivantes :

1. Elles partent au C.V au début de la journée. Autrement dit, elles doivent commencer leur premier déplacement de la journée à partir d'endroits situés hors du C.V et se diriger vers des endroits situés dans le C.V.
2. L'une des deux personnes se déplace vers le C.V en mode voiture.

## Chapitre V. Mise en oeuvre et expérimentation du TST

---

3. Les deux personnes partent vers le centre ville et une partie de leur itinéraire est spatialement proche l'une de l'autre.
4. Les deux personnes doivent partir à des horaires avoisinantes.

Pour formuler cette requête, nous avons eu recours aux différents opérateurs sémantiques, spatiaux, temporels et spatio-temporaux de notre langage. En effet, pour étudier le comportement de ces personnes, seul le premier déplacement est sollicité. Nous appliquons donc nos différents opérateurs, principalement, sur la restriction des TSTs au premier couple activité/déplacement et ceci toujours à l'aide de l'opération *At\_Activity\_Activity*. La première condition est vérifiée par l'utilisation de l'opération spatiale du SGBD PostGIS "*contains*" qui vérifie qu'une géométrie est bien à l'intérieur d'une autre. Il suffit alors de vérifier que les premiers et les derniers lieux (récupérés par les opérations sur le TST : *First\_Point* et *Last\_Point*) des sous-trajectoires résultats de la restriction sont bien contenus dans le C.V. La deuxième condition est vérifiée en s'assurant que le nombre de déplacements en mode voiture dans les sous-trajectoires est soit égal à 0 ou à 1. Notre opération de voisinage spatial, *TST\_SpatialNeighbour\_TST*, entre TST, est appliquée aux sous trajectoires pour vérifier si les deux personnes ont parcourus des itinéraires vers le C.V qui sont proches spatialement. Cette proximité est fixée par un seuil qui pourra être défini et donné par l'utilisateur lors de la formulation de la requête. Finalement, pour vérifier le "voisinage" temporel des temps de départ des deux déplacements, nous calculons la différence entre les temps de départ des deux trajectoires et nous le comparons à un seuil qui peut être pris égale à 10 minutes par exemple.

```
SELECT  DISTINCT pr1.id , pr2.id,First_Time(pr1.traj), First_Time(pr2.traj),
        (First_Time(pr1.traj)::time - First_Time(pr2.traj)::time)
FROM    municipalites as cv,
        (SELECT id, traj, At_Activity_Activity(pr.traj, First_Activity(pr.traj),
        First_Activity(pr.traj)) AS sousTST1
FROM    personne as pr , municipalites as cv
WHERE   length(pr.traj) >= 1
AND     cv.nom_mun = 'Charlesbourg'
--partent d'endroit hors C.V
AND     not contains(cv.zone,First_Point(pr.traj))
AND     contains(cv.zone,Last_Point(At_Activity_Activity(pr.traj,
        First_Activity(pr.traj),First_Activity(pr.traj)))
        ) --partent vers des endroits au C.V
AND     Trips_With_Mode_Count(At_Activity_Activity(pr.traj,
```

```

                                First_Activity(pr.traj),First_Activity(pr.traj)
                                ), '1') = 0 -- partent sans voiture
) AS pr1,

(SELECT id, traj, At_Activity_Activity(pr.traj, First_Activity(pr.traj),
                                First_Activity(pr.traj)) AS sousTST2
FROM personne as pr , municipalites as cv
WHERE length(pr.traj) >= 1
      AND cv.nom_mun = 'Charlesbourg'
      AND not contains(cv.zone,First_Point(pr.traj))
      AND contains(cv.zone,Last_Point(At_Activity_Activity(pr.traj,
                                First_Activity(pr.traj),First_Activity(pr.traj))))
      AND Trips_With_Mode_Count(At_Activity_Activity(pr.traj,
                                First_Activity(pr.traj),First_Activity(pr.traj))
                                , '1') <> 0 -- partent avec voiture
) AS pr2

WHERE pr1.id <> pr2.id
AND TST_SpatialNeighbour_TST(pr1.sousTST1,pr2.sousTST2, 1000)
AND(
  (First_Time(pr2.traj)::time <= First_Time(pr1.traj)::time And
  (First_Time(pr1.traj)::time -
                                First_Time(pr2.traj)::time) < '00:10:00'::interval)
  OR
  ((First_Time(pr2.traj)::time >= First_Time(pr1.traj)::time
  And (First_Time(pr2.traj)::time -
                                First_Time(pr1.traj)::time) < '00:10:00'::interval))
)
order by pr1.id;

```

La figure [V.13](#) montre, sous l'outil QGIS, les itinéraires des couples de personnes répondant à cette requête et donc pouvant faire du co-voiturage pour aller au C.V au début de la journée. Dans cette image, les points de départ (hors du C.V) des différentes personnes concernées sont mentionnées en bleu alors que les points d'arrivée (dans le C.V) sont mentionnés en vert. Les itinéraires entre ces points sont indiqués en rouge.

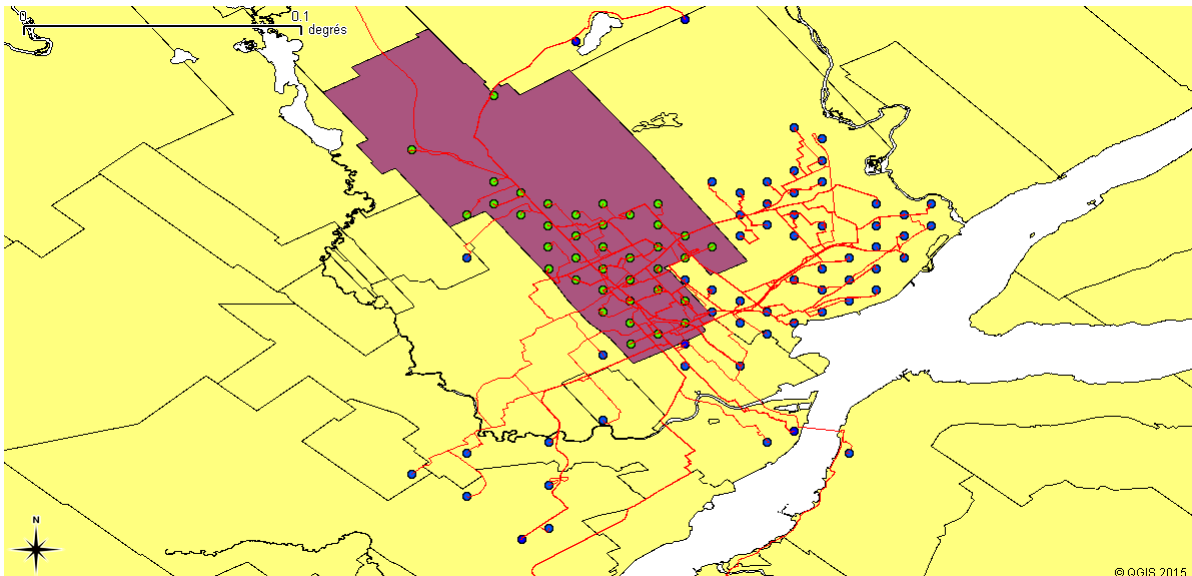


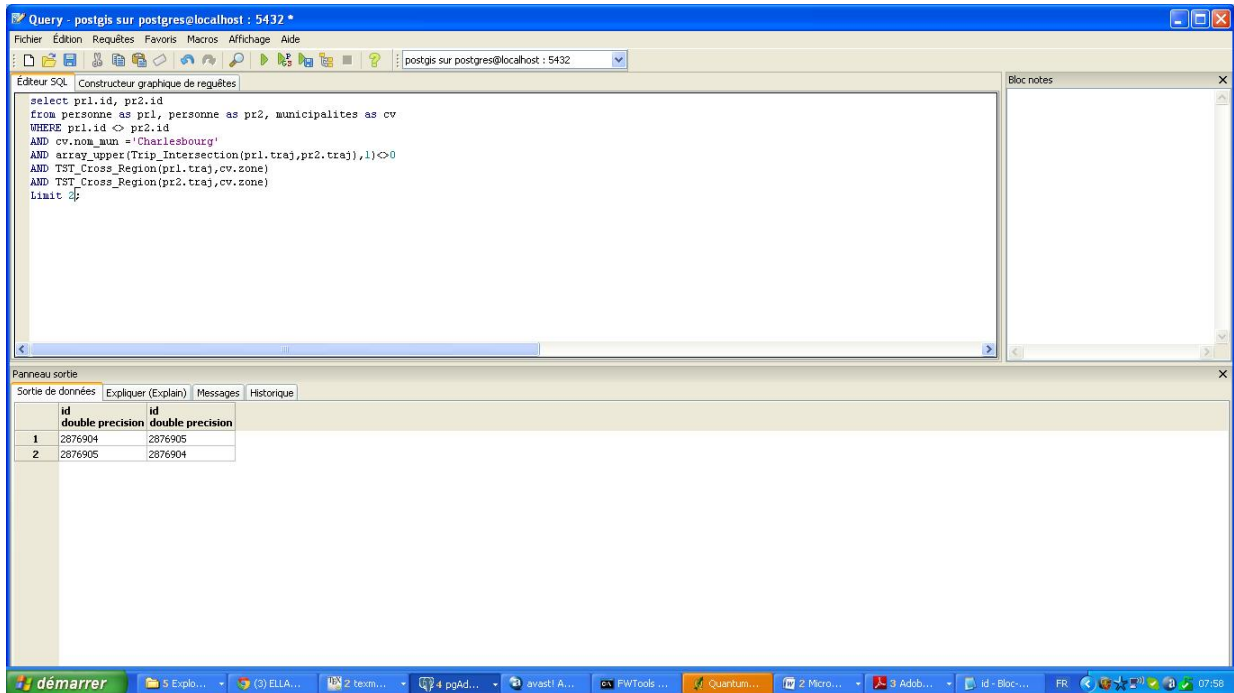
Figure V.13 – Visualisation des résultats de la requête 7

**Requête 8 :** Quelles sont les paires de personnes qui sont passées par le C.V et qui ont des déplacements en commun?

Ce genre de requête peut être complémentaire à la requête 7. En effet, le résultat de cette requête est la paire de personnes ayant des itinéraires communs, démarrant aux mêmes instants de temps de départ et arrivant aux mêmes temps d'arrivée et utilisant les mêmes modes de déplacements pour faire la même activité. Ces paires de personnes, peuvent représenter des personnes de la même famille par exemple un couple de mari et femme, ou des collègues de travail allant déjeuner ensemble après le travail. Durant ces déplacements, ils devraient faire du co-voiturage ou prendre le même taxi au lieu d'utiliser chacun un mode de transport différent. Pour avoir comme résultat des paires de personnes qui vérifie cette condition, il nous suffit d'appliquer l'opération *Trips\_Intersection* entre les attributs TST des personnes et de tester que le résultat obtenu est une liste de déplacements qui n'est pas vide. La fonction *TST\_Cross\_Region* nous permet de tester si ces personnes sont bien passées par le C.V ou pas.

```
select pr1.id, pr2.id
from personne as pr1, personne as pr2, municipalites as cv
WHERE pr1.id <> pr2.id
AND array_upper(Trip_Intersection(pr1.traj,pr2.traj),1)<>0
AND TST_Cross_Region(pr1.traj,cv.zone) AND TST_Cross_Region(pr2.traj,cv.zone);
```

Le résultats de cette requête, limité à deux enregistrements seulement pour que ça soit plus



**Figure V.14** – Résultat de la requête 8 limité à deux enregistrements

visible, est donné dans la figure V.14. La paire de personnes retournée sont des personnes de la même famille qui ont eu un déplacement en commun.

La figure V.15 montre les itinéraires de ces deux personnes. Les trajectoires sont égales vu que les deux personnes ont eu exactement le même comportement de déplacement.

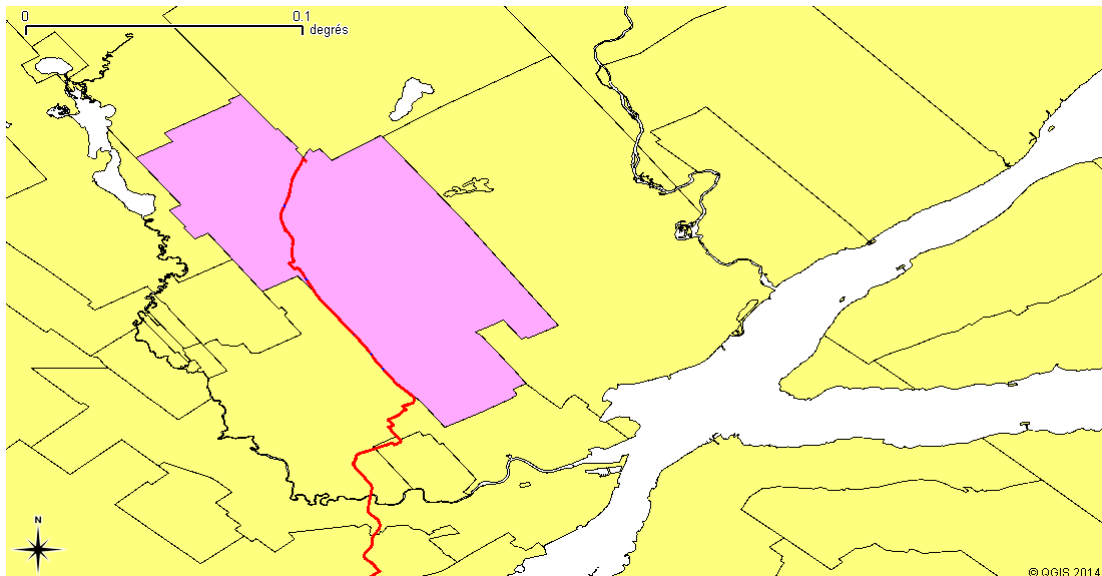
## 4 Positionnement du modèle proposé

Nous avons présenté dans le chapitre état de l'art une étude comparative des modèles de trajectoires spatio-temporelles que nous avons étudiés. Nous avons remarqué que peu de modèles existant couvrent la sémantique du mouvement liée aux besoins des applications tels que les informations relatives au comment et au pourquoi des déplacements ainsi qu'à la représentation explicite des aspects statiques et dynamiques du mouvement et des relations entre eux.

Les modèles qui traitent cet aspect sont caractérisés soit par un manque de généralité et sont donc liés à une application bien spécifique [6] soit par une pauvreté dans la sémantique qu'ils modélisent [37], soit par une insuffisance, voire une absence des mécanismes de manipulations de la trajectoire qu'ils proposent [29, 76].

Notre modèle permet de couvrir l'aspect sémantique désiré grâce à l'adoption du cadre de Time Geography et de sa conception de la trajectoire spatio-temporelle comme un chaînage





**Figure V.15** – Visualisation du résultat de la requête 8

d'activités et de déplacements liés sémantiquement. Le modèle proposé dans cette thèse est indépendant d'un type particulier d'applications et est donc ouvert à un très grand large ensemble d'applications gérant des objets mobiles.

D'autre part, peu de modèles, existant dans la littérature et s'appuyant sur le cadre de Time Geography, présentent une formalisation rigoureuse de ses concepts clés et offrent une capacité riche et efficace de ces concepts. À l'aide d'une modélisation par type abstrait algébrique, nous avons pu offrir une formalisation mathématique, donc rigoureuse et indépendante d'un type particulier de SGBD, des différentes composantes qui peuvent constituer une trajectoire spatio-temporelle sémantiquement riche. Nous avons pu, aussi, offrir aux utilisateurs des bases de données spatio-temporelles un nouveau type qui intègre d'une façon cohérente et homogène ses notions et langage de manipulation riche permettant leur manipulation.

D'ailleurs, à notre connaissance, aucun des modèles proposés n'offre un pouvoir de manipulation de la trajectoire spatio-temporelle modélisée à différentes dimensions, à la fois sémantique, spatiale, temporelle et spatio-temporelle. Notre modèle offre aussi des mécanismes qui permettent de garantir l'intégrité des données trajectoires au sein des bases de données spatio-temporelles. Seul [57] offre de tels mécanismes.

Enfin, nous pouvons dire que notre modèle représente, une contribution qui se positionne donc à l'intersection du cadre de Time Geography et du domaine des bases de données spatio-temporelles. Peu de modèles de la trajectoires spatio-temporelles étudiés dans la littérature le sont.

## 5 Conclusion

La mise en oeuvre de notre modèle consiste à l'implémenter au sein d'un SGBD spatial en tant que type de données et de le mettre ainsi à la disposition des utilisateurs de bases de données spatio-temporelles.

Notre choix de SGBD s'est porté sur le SGBD spatial open source PostGIS au sein duquel nous avons intégré le type TST en tant que type composite. Cette implémentation nous a suffi pour démontrer le potentiel d'utilisation et le pouvoir d'expression de notre langage de requête, dans un contexte d'étude de la mobilité au centre ville.

En effet, cet exemple d'application illustre le potentiel de notre langage dans le cadre de l'interrogation de la modélisation et la manipulation de mobilité de personnes. Un des avantages du modèle et du langage proposés est qu'il permet d'intégrer les propriétés spatiales, temporelles spatio-temporelles et sémantiques des opérateurs au sein d'une approche de modélisation unifiée. La sémantique spécifique des trajectoires spatio-temporelles se traduit par une intégration des concepts de déplacements et d'activités. C'est la conjonction de ces différentes catégories d'opérations qui constitue le principal intérêt du langage proposé.

D'autre part, la mise en oeuvre de notre langage en tant que type de base de PostGIS doit être réalisée à fin de garantir une intégration plus robuste du type abstrait de données. Des tests supplémentaires, sur des données comportant des trajectoires réelles, permettrait une meilleure interprétation des tests effectués.

---

## Conclusion

Avec le développement continu des technologies de capteurs de données en temps réel, et donc du potentiel lié à la modélisation d'objets mobiles, la communauté de recherche en bases de données spatio-temporelles s'est largement préoccupée ces dernières années du développement de modèles conceptuels ou formels de représentation des trajectoires d'objets mobiles. Les problématiques abordées dans ces recherches ont été celles de la représentation de la continuité du mouvement ou la prise en considération d'objets géométriques mobiles plutôt que de simples objets ponctuels mobiles.

D'autres démarches ont abordé des caractérisations sémantiques des notions de trajectoires pour des finalités d'optimisation des capacités de stockage et pour une meilleure compréhension, gestion et exploitation du comportement de mobilité des objets.

La majorité de ces propositions se sont limitées à une certaine forme de sémantique, dérivée de données brutes de trajectoires (e.g le changement de vitesse des objets) et ont négligé de prendre en considération une forme de sémantique liée aux besoins des applications (e.g associer une connaissance liée au pourquoi et au comment du déplacement).

Dans le contexte spécifique d'études sociales sur la mobilité individuelle de personnes dans un environnement urbain, le cadre conceptuel de Time Geography offre dans sa vision de la trajectoire spatio-temporelle une dimension sémantique qui pourrait être généralisée à l'étude de différents types d'objets mobiles. Cette forme de sémantique, émanant d'une représentation des trajectoires sous la forme d'événements reliés sémantiquement entre eux, répond aux besoins d'analyse et de compréhension de la mobilité pour un large ensemble d'applications gérant des objets mobiles. Néanmoins, le cadre de Time Geography a été peu exploité et ses notions clés peu formalisées dans les représentations des trajectoires spatio-temporelles au sein de bases de données spatio-temporelles en général et de bases de données d'objets mobiles en particulier.

Le constat qui découle des ces différentes problématiques est alors que les notions de trajectoires restent encore à intégrer au sein d'approches à la fois formelles et sémantiques qui permettent leur représentation et manipulation au sein des bases de données gérant des objets mobiles.

## Contribution

Cette thèse vient compléter les représentations existantes au sein des bases de données spatio-temporelles en proposant une modélisation algébrique des trajectoires spatio-temporelles individuelles d'objets mobiles, exprimant des déplacements et des activités reliés sémantiquement. Ces notions de déplacements et d'activités jouent ici un rôle particulier dans la construction du modèle, rôle qui se fonde sur les principes souvent utilisés dans le cadre de Time Geography.

La modélisation proposée offre d'une part, une représentation sémantiquement riche des trajectoires sous la forme d'un nouveau type abstrait de données, nommé TST, et d'autre part, des fonctions de manipulation à des dimensions spatiales, temporelles et spatio-temporelles ainsi qu'à une dimension sémantique.

Le choix du cadre de Time Geography, peut être considéré en lui même comme une contribution, dans la mesure où la notion de trajectoire spatio-temporelle énoncée dans ce cadre, (*space-time path*), a été très peu formalisée jusqu'ici dans la littérature et que notre modèle utilise une approche formelle pour la représentation de cette notion.

Après un état de l'art où différents modèles de trajectoires spatio-temporelles ont été étudiés, selon des critères que nous avons fixés au préalable, une synthèse a été dressée. Cette synthèse a montré une insuffisance de ces modèles à couvrir les besoins d'intégration et de manipulation de la sémantique, telle que nous la percevons, et à assurer en même temps une représentation unifiée de toutes les caractéristiques spatiales, temporelles et sémantiques des trajectoires d'objets mobiles.

Avant d'attaquer la proposition du TST, et pour pouvoir couvrir un maximum d'aspects opérationnels nécessaires lors de la manipulation de ce type, nous avons entamé une étude sur les besoins d'analyse des trajectoires spatio-temporelles. Cette étude a permis d'identifier différentes dimensions d'analyse. Pour chaque dimension, nous avons dégagé différents sortes de manipulations de la trajectoire spatio-temporelle.

Sur la base de cette étude, nous avons proposé une modélisation algébrique du type TST en deux étapes :

1. d'abord, une spécification formelle dénotationnelle du TST est donnée. Cette spécification représente la signature ou le modèle syntaxique du type proposé,
2. ensuite l'algèbre liée à cette signature est présentée. L'algèbre proposée représente le modèle sémantique du type. Autrement dit, elle définit les données constituant le TST et exprime ses opérations de manipulations à l'aide de formalisations mathématiques offrant une rigueur et une indépendance dans la représentation d'un type particulier de SGBD.

Cette approche par type abstrait de données nous a permis de considérer un TST comme une entité identifiable exprimant des composantes spatiales, temporelles ainsi qu'une composante sémantique de la trajectoire et de proposer un langage de manipulation de ces différentes composantes. Le langage de manipulation permet de manipuler le concept de trajectoire à partir d'un concept homogène de représentation, et où les différents opérateurs identifiés peuvent se combiner et ainsi offrir un fort potentiel de manipulation. Le potentiel du TST est relativement large, il doit permettre de modéliser des phénomènes multiples de déplacement et d'activités, pour un grand nombre d'applications pour les sciences environnementales et sociales, aux sciences de l'ingénieur, que les dynamiques représentées concernent des humains ou tout type d'objets mobiles.

Nous avons proposé un ensemble d'algorithmes qui décrivent le comportement des opérations du TST. Une mise en oeuvre du type proposé et du langage de manipulation qui lui est associé est proposée. Nous avons en effet intégré le type TST, en tant que type composite, et ses différentes opérations au sein du SGBD spatial open source PostgreSQL/PostGIS.

La validation de ce travail a pour objectif de décrire le potentiel et les possibilités de notre modèle de représentation et de manipulation de trajectoires dans un contexte d'étude. Dans ce cas, le choix s'est porté sur le cadre d'une application d'étude de la problématique de la mobilité au centre ville. Des données benchmark issues d'une enquête Origine/Destination, conduite en 2001 par la Société de Transport de la Communauté Urbaine de Québec (STCUQ), sont traités puis utilisés comme jeu de test.

L'approche que nous utilisons pour la validation est la proposition d'un lot de requêtes à différentes dimensions, spatiales, temporelles, sémantiques et spatio-temporelles et la formulation de ces requêtes en utilisant les langages de requête. L'intérêt du type TST et de ses opérations apparaît dans plusieurs cas de requêtes, spécialement de nature sémantique, et dans d'autres cas où différents opérateurs du TST peuvent se combiner et ainsi produire des requêtes à valeur ajoutée.

De part les apports que nous avons cités, cette thèse constitue une contribution aux recherches en cours dans les domaines de bases de données spatio-temporelles et de l'ingénierie

de l'information géographique en général et des SIG en particulier. Cependant, nous avons pu déceler quelques manques dans notre modélisation relatifs à l'aspect continuité du mouvement qui n'a pas été pris en considération dans cette thèse. Autrement dit, la modélisation du mouvement que nous proposons a un côté discontinu. Autrement dit, les informations spatiales modélisées au sein d'un TST sont de natures géométriques (des points et des polygones). Cette représentation est suffisante pour certaines applications. Une représentation continue du mouvement rendrait le type TST encore plus riche.

D'autre part, nous avons supposé, dans notre proposition, la connaissance des données sémantiques et nous avons, juste, proposé un modèle pour sa représentation et sa manipulation au sein des trajectoires spatio-temporelles. Le problème de l'acquisition de cette connaissance sémantique reste un point à traiter pour compléter notre travail.

## Perspectives de recherche

Suite aux différents manques présentés plus haut, plusieurs perspectives de notre travail peuvent être envisagées. Ces perspectives peuvent être relatives soit au modèle proposé, soit à sa mise en oeuvre.

### Du point de vue de modèle

Le modèle que nous proposons ne prends pas en considération l'aspect continu du déplacement. En effet, l'itinéraire du mouvement est représenté par une information géométrique de type ligne. Il serait très intéressant de prendre en considération cet aspect continue pour compléter les informations spatio-temporelles au sein du TST et avoir donc de meilleurs résultats de manipulation. Ainsi, une autre approche pourrait être adoptée pour illustrer la continuité temporelle. Dans ce cas, l'itinéraire (*Path*) d'un déplacement pourra être représenté, par exemple, par le type *mpoint* proposé dans [71] et d'autres formes de requêtes, principalement, spatio-temporelle, pourraient être formulées sur la base de l'interface du type *mpoint*. Sur cette base pourraient être modélisées des requêtes sur la position exacte d'un individu à un instant ou intervalle donnés.

D'autre part, nous assimilons dans notre recherche l'objet mobile manipulé à un point mobile. Afin de couvrir une représentation encore plus grande de phénomènes, on pourrait imaginer d'étendre cette représentation à la manipulation d'un ensemble de points mobiles, représentant par exemple une horde d'oiseaux migrateurs ou de régions mobiles, représentant, par exemple, des phénomènes météorologiques comme les ouragans.

Le modèle proposé dispose des informations nécessaires à l'utilisation d'autres concepts de la Time Geography comme les notions de *contraintes*, de *prisme spatio-temporel* ou le concept de *poches d'ordre local*. Dans un contexte moins formel, et à travers le concept de prisme spatio-temporel, le langage pourrait offrir dans une étape ultérieure de nouvelles possibilités de manipulations visuelles du type TST. Les poches d'ordre local (ou *Bundle*) qui sont liés aux regroupements de trajectoires pour l'exercice d'activités communes peuvent aussi être identifiées par visualisation en trois dimensions des TST. Une extension de ce modèle pourrait permettre de dégager formellement ces agrégations.

L'information sémantique dans notre approche est supposée connue, à titre d'exemple elle pourrait être introduite par l'utilisateur ou extraite manuellement dans une phase de pré-traitement de la base de données. Il serait très utile et intéressant de proposer une approche d'extraction de la sémantique du mouvement à partir de données brutes, comme des trajectoires brutes ou d'autres données comportant cette sémantique comme des enquêtes Origine-Destination.

D'autre part, nous avons proposé notre modèle en partant du fait que l'échelle spatiale et temporelle sont fixées dès le départ par l'utilisateur de l'application. Il serait intéressant d'offrir à l'utilisateur la possibilité de manipuler le type TST sur plusieurs échelles en même temps ; autrement dit, de proposer une représentation multi-échelles du type TST.

## Du point de vue de la mise en oeuvre

Nous nous sommes servis de la classe de type, "type composite", pour implémenter notre type TST au sein du SGBD PostgreSQL/PostGIS et effectuer un prototypage aisé et rapide. Comme prochaine étape, il faudrait intégrer le type TST, et ses différentes opérations de manipulation, au sein du noyau du SGBD comme un type de base. Ceci doit se faire à l'aide d'un langage bas niveau comme le langage C.

Afin d'apporter plus de facilité et d'interaction à la manipulation du type TST, il reste à développer un prototype comportant une interface graphique qui permet une manipulation et un requêtage d'une base de données spatio-temporelles moyennant le langage que nous proposons. L'interface, pourra à titre d'exemple, offrir aux utilisateurs, une manipulation selon une dimension particulière ou plusieurs dimensions combinées de la trajectoire spatio-temporelle.

Afin d'avoir une visualisation des trajectoires, il serait intéressant de permettre un accès direct du prototype à un logiciel SIG afin de bénéficier d'une vision cartographique et des fonctionnalités de ce dernier. Une animation cartographique permettrait, aussi, de procurer des moyens supplémentaires pour illustrer les composantes spatiales des trajectoires spatio-

temporelles d'objets mobiles. La visualisation pourrait se faire en trois dimensions et une annotation sémantique (natures des activités, modes,...) des trajectoires pourrait être rajoutée. En effet, cette approche de visualisation tri-dimensionnelle permettra d'offrir une visualisation de l'espace-temps plus fidèle à la nature même des phénomènes [4] et permettra l'exploitation plus tard des concepts du Time Geography, tels que les prismes spatio-temporels ou les poches d'ordre local .



---

---

# Annexe A

---

## Algorithmes proposés

---

### Algorithme 1 Make

---

/\* Associé à la fonction d'interprétation (IV.12) \*/  
**Sorties** : un objet  $tst$  de type  $TST$   
 $tst := (\emptyset, \emptyset);$   
**retourner**  $tst;$

---

---

### Algorithme 2 Add

---

/\* Associé à la fonction d'interprétation (IV.13) \*/  
**Entrées** : un objet  $tst$  de type  $TST$ , un objet  $a$  de type  $Activity$  et un objet  $d$  de type  $Trip$   
**Sorties** : un objet  $tst$  de type  $TST$   
Soit  $\alpha$  une unité de temps de l'échelle temporelle définie par l'utilisateur  
**si**  $check\_activity(a) \neq true$  **alors**  
     $ecrire('erreur : l'activité donnée n'est pas valide');$   
**sinon**  
    **si**  $check\_trip(d) \neq true$  **alors**  
         $ecrire('erreur : le déplacement donné n'est pas valide');$   
    **sinon**  
        **si**  $check\_tst(tst) \neq true$  **alors**  
             $ecrire('erreur : le TST donné n'est pas valide');$   
        **sinon**  
            **si**  $d.l_s = a_n.l \wedge d.l_e = a.l \wedge d.t_s = a_n.t_e + \alpha \wedge d.t_e = a.t_s - \alpha$  **alors**  
                 $tst := (tst.A \cup \{a\}, tst.d \cup \{d\});$   
            **fin si**  
        **fin si**  
    **fin si**  
**retourner**  $tst$

---

---

**Algorithme 3 Activities\_With\_Mode\_Count**

---

/\* Associé à la fonction d'interprétation (IV.17) \*/

**Entrées** : un objet *tst* de type *TST*, un objet *m* de type *string***Sorties** : un entier *count* de type *integer*vérifier la validité de *tst* avec *check<sub>t</sub>st*, si c'est le cas parcourir l'ensemble de ces déplacements :*count* := 0;**pour** chaque  $d_i \in \text{tst}.D$  **faire**  **si**  $m \in d_i.\text{mode}$  **alors**    *count* := *count* + 1;  **fin si****fin pour****retourner** *count*;

---

**Algorithme 4 Nth\_Activity**

---

/\* Associé à la fonction d'interprétation (IV.25) \*/

**Entrées** : un objet *tst* de type *TST*, un entier *i* de type *integer***Sorties** : un objet *a* de type *Activity*vérifier la validité de *tst*, si c'est le cas :*n* := *card*(*tst.A*);**si**  $n = 0$  **alors**  *ecrire*('le TST est vide');**sinon**  **si**  $i < 1 \wedge i > n$  **alors**    *write*('La position donnée en argument n'existe pas dans le TST');  **sinon****retourner** *tst.A.a<sub>i</sub>*;  **fin si****fin si**

---

**Algorithme 5 Position\_Activity**

---

/\* Associé à la fonction d'interprétation (IV.43) \*/

**Entrées** : un objet *tst* de type *TST*, un objet *a* de type *Activity***Sorties** : un entier *pos* de type *Integer*Vérifier la validité de *tst* et de *a*, si c'est le cas :*pos* := 0;**pour** chaque activité *a<sub>i</sub>* du *tst* **faire**  **si**  $a_i = a$  **alors**    **Retourner** *i*  **fin si****fin pour****retourner** *pos*

---

**Algorithme 6 Activity\_Before\_Activity**

---

/\* Associé à la fonction d'interprétation (IV.33) \*/

**Entrées** : un objet *tst* de type *TST*, un objet *a* de type *Activity*

**Sorties** : un objet *ares* de type *Activity*

vérifier la validité de *tst* et de *a*, si c'est le cas :

$n := \text{card}(tst.A)$

**si**  $n \neq 0$  **alors**

**si**  $\text{Include\_Activity}(tst, a) = \text{true}$  **alors**

$pos := \text{Position\_Activity}(tst, a)$ ;

**si**  $pos \neq 1$  **alors**

$ares := tst.A.a_{pos-1}$ ;

**Return** *ares*;

**sinon**

**ecrire**('L'activité en argument est la première dans le TST, il n'existe pas d'activités qui la précèdent');

**fin si**

**sinon**

**ecrire**('L'activité en argument n'existe pas dans le TST');

**fin si**

**sinon**

**ecrire**('TST vide');

**fin si**

---

---

**Algorithme 7 Activities\_Before\_Activity**

---

/\* Associé à la fonction d'interprétation (IV.41) \*/

**Entrées** : un objet *tst* de type *TST*, un objet *a* de type *Activity***Sorties** : une liste d'activités *alist* de type *Alist*vérifier la validité de *tst* et de *a*, si c'est le cas :*n* := *card*(*tst.A*);*alist* :=  $\emptyset$ ;**si** *n*  $\neq$  0 **alors**  **si** *Include\_Activity*(*tst, a*) = *true* **alors**    *pos* := *Position\_Activity*(*tst, a*);    **si** *pos*  $\leq$  1 **alors**      *list* := *alist*  $\cup$  {*a<sub>i</sub>*  $\in$  *tst1.A* | 1  $\leq$  *i*  $\leq$  *pos*}      **Return** *alist*;    **sinon**      *ecrire*('L'activité en argument est la première dans le TST, il n'existe pas d'activités qui la précèdent');    **fin si**  **sinon**    *ecrire*('L'activité en argument n'existe pas dans le TST');  **fin si****sinon**  *ecrire*('TST vide');**fin si**

---

---

**Algorithme 8 Activities\_With\_Mode**

---

/\* Associé à la fonction d'interprétation (IV.48) \*/

**Entrées** : un objet *tst* de type *TST*, un objet *m* de type *string***Sorties** : une liste d'activités *alist* de type *Alist*vérifier la validité de *tst* avec *check<sub>t</sub>st*, si c'est le cas parcourir l'ensemble de ces déplacements :*alist* :=  $\emptyset$ ;**pour** chaque *d<sub>i</sub>*  $\in$  *tst.D* **faire**  **si** *m*  $\in$  *d<sub>i</sub>.mode* **alors**    *alist* := *alist*  $\cup$  {*a<sub>i</sub>*};  **fin si****fin pour****retourner** *alist*;

---

---

**Algorithme 9 First\_Activity\_With\_Type**

---

/\* Associé à la fonction d'interprétation (IV.49) \*/

**Entrées** : un objet  $tst$  de type  $TST$ , un objet  $m$  de type  $String$

**Sorties** : un objet  $a$  de type  $Activity$

vérifier la validité de  $tst$  avec  $check_{tst}$ , si c'est le cas parcourir l'ensemble de ces activités :

$n = card(tst.A)$ ;

**si**  $n = 0$  **alors**

    écrire('le TST est vide');

**sinon**

**pour** chaque  $a_i \in tst.A$  en commençant le parcours par  $i = 1$  **faire**

**si**  $a_i.type = m$  **alors**

**Retourner**  $a_i$

**fin si**

**fin pour**

    écrire('il n'existe aucune activité ayant ce type');

**fin si**

---

---

**Algorithme 10 Next\_Activity\_With\_Type**

---

/\* Associé à la fonction d'interprétation (IV.51) \*/

**Entrées** : un objet *tst* de type *TST*, un objet *a* de type *Activity* et un objet *m* de type *String***Sorties** : un objet *ares* de type *Activity*vérifier la validité de *tst* et de *a*, si c'est le cas parcourir l'ensemble des activités de *tst* :*n* := *card*(*tst.A*);**si** *n* ≠ 0 **alors**    *i* := *Position\_Activity*(*tst*, *a*)    **si** *i* ≠ 0 **alors**        **si** *i* ≠ *n* **alors**            **pour** chaque  $a_i \in \{a_i \in \textit{tst.A} \mid \textit{pos} + 1 \leq i \leq n\}$  **faire**                **si**  $a_i.type = m$  **alors**                    **Retourner**  $a_i$                 **fin si**            **fin pour**            *ecire*('Aucune activité ayant le type indiqué ne suit l'activité en argument')    **sinon**        *ecire*('L'activité en argument est la dernière dans le TST elle ne possède pas de suivant');    **fin si**    **sinon**        *ecire*('L'activité en argument n'existe pas dans le TST');    **fin si****sinon**    *ecire*('TST vide')**fin si**

---

---

**Algorithme 11 Previous\_Activity\_With\_Type**

---

/\* Associé à la fonction d'interprétation (IV.52) \*/

**Entrées :** un objet *tst* de type *TST*, un objet *a* de type *Activity* et un objet *m* de type *String*

**Sorties :** un objet *ares* de type *Activity*

vérifier la validité de *tst* et de *a*, si c'est le cas parcourir l'ensemble des activités de *tst* :

$n := \text{card}(tst.A)$ ;

**si**  $n \neq 0$  **alors**

$i := \text{Position\_Activity}(tst, a)$

**si**  $i \neq 0$  **alors**

**si**  $i \neq 1$  **alors**

**pour** chaque  $a_i \in \{a_i \in tst.A \mid i \leq i \leq pos - 1\}$  **aller** de  $pos-1$  à  $i$  **faire**

**si**  $a_i.type = m$  **alors**

**Retourner**  $a_i$

**fin si**

**fin pour**

**ecrire**('Aucune activité ayant le type indiqué ne précède l'activité en argument')

**sinon**

**ecrire**('L'activité en argument est la première dans le TST elle ne possède pas de précédent');

**fin si**

**sinon**

**ecrire**('L'activité en argument n'existe pas dans le TST');

**fin si**

**sinon**

**ecrire**('TST vide')

**fin si**

---

---

**Algorithme 12** *At\_Activity\_Activity*

---

/\* Associé à la fonction d'interprétation (IV.65) \*/

**Entrées** : un objet *tst* de type *TST*, deux activités *x* et *y* de type *Activity***Sorties** : un objet *res* de type *TST*vérifier la validité de *tst* et des activités *x* et *y*, si c'est le cas :*res* := *make*();*n* := *card*(*tst.A*);**si** *n* = 0 **alors**    *k* := *Position\_Activity*(*tst*, *x*);    *j* := *Position\_Activity*(*tst*, *y*);

{ Si l'activité a1 précède l'activité a2 dans le TST }

**si** *k* <> 0 ou *j* <> 0 **alors**        **si** *k* ≤ *j* **alors**            **pour** chaque activité  $a_i \in \{a_i \in \textit{tst.A} \mid k \leq i \leq j\}$  **faire**                *res* := *Add*(*tst\_res*, *tst.A.a<sub>i</sub>*, *tst.A.d<sub>i</sub>*);            **fin pour**            **Retourner** *res*        **sinon**            *ecrire*('La première activité en argument doit précéder dans le TST la deuxième activité en argument')        **fin si**    **sinon**        *ecrire*('L'une des activités en arguments n'existe pas dans le TST')    **fin si****sinon**    *ecrire*('Le TST en argument est vide')**fin si**

---



---

**Algorithme 13 Union**

---

/\* Associé à la fonction d'interprétation (IV.170) \*/

**Entrées :** deux objets *tst1* et *tst2* de type *TST*

**Sorties :** un objet *res* de type *TST* représentant le premier *tst* résultat de l'intersection de *tst1* et *tst2*

vérifier la validité de *tst1* et *tst2*, si c'est le cas :

*res* := Make();

**si** *Equals(tst1, tst2)* **ou** *Includes(tst2, tst1)* **alors**

*res* := *tst1*;

**sinon**

**si** *Includes(tst1, tst2)* **alors**

*re* := *tst2*;

**sinon**

**si** (*tst2.D.d1.l<sub>s</sub>* = *tst1.A.a<sub>n</sub>.l*) **et** (*tst2.D.d1.t<sub>s</sub>* = *tst1.A.a<sub>n</sub>.t<sub>e</sub>* + 1) **ou** ((*tst1.D.d1.l<sub>s</sub>* = *tst2.A.a<sub>n</sub>.l*) **et** (*tst1.D.d1.t<sub>s</sub>* = *tst2.A.a<sub>n</sub>.t<sub>e</sub>* + 1)) **alors**

*res* := (*tst1.A* ∪ *tst2.A*, *tst1.D* ∪ *tst2.D*)

**sinon**

**si** les deux trajectoires *tst1* et *tst2* possèdent chacun un ensemble d'activités tels que  $A1 = \{a_i, a_{i+1}, \dots, a_{n1}\} \in \text{tst1.A}$  et  $A2 = \{a_1, a_2, \dots, a_k\} \in \text{tst2.A}$  et  $A1 = A2$  **alors**

                /\* Concaténer à la fin de *tst1* la sous trajectoire de *tst2* commençant par l'activité  $a_{k+1}$  : \*/

*res* := Concaténer\_bout\_commun(*tst1*, *tst2*, *k*)

**sinon**

**si** les deux trajectoires *tst1* et *tst2* possèdent chacun un ensemble d'activités tels que  $A1 = \{a_i, a_{i+1}, \dots, a_{n1}\} \in \text{tst2.A}$  et  $A2 = \{a_1, a_2, \dots, a_k\} \in \text{tst1.A}$  et  $A1 = A2$  **alors**

                /\* Concaténer à la fin de *tst2* la sous trajectoire de *tst1* commençant par l'activité  $a_{k+1}$  : \*/

*res* := Concaténer\_bout\_commun(*tst2*, *tst1*, *k*)

**fin si**

**fin si**

**fin si**

**fin si**

**fin si**

retourner *res*;

---

---

**Algorithme 14 Concaténer\_bout\_commun**

---

**Entrées :** deux objets  $tst\_deb$  et  $tst\_fin$  de type  $TST$  et un entier  $k$  de type  $Integer$

**Sorties :** un objet  $res$  de type  $TST$  résultat de la concaténation de  $tst2$  à la fin de  $tst1$   
vérifier la validité de  $tst\_deb$  et  $tst\_fin$ , si c'est le cas :

$res := Make();$

$tst\_temp := Make();$

$tst\_temp := At\_Activity(tst\_fin, Next\_Activity(tst\_fin, tst\_fin.a_{k+}), tst\_fin.a_n);$

$res := (tst\_deb.A \cup tst\_temp.A, tst\_deb.D \cup tst\_temp.D);$

**retourner**  $res$ ;

---

---

**Algorithme 15 Intersection**

---

*/\* Associé à la fonction d'interprétation (IV.166) \*/*

vérifier la validité de  $tst1$  et  $tst2$ , si c'est le cas :  $res := Make();$

parcourir l'ensemble des activités de  $tst1$  et de  $tst2$  et retrouver les positions  $j$  dans  $tst1$  et  $k$  dans  $tst2$  de la première activité commune.

**si**  $j \neq 0$  et  $k \neq 0$  **alors**

**tant que**  $j \leq card(tst1)$  et  $k \leq card(tst2)$  et  $tst1.A.a_j = tst2.A.a_k$  **faire**

$res := Add(res, tst1.A.a_j, tst1.D.d_j);$

$j := j + 1;$

**fin tant que**

**fin si**

**retourner**  $res$ ;

---

---

**Algorithme 16 Activities\_Intersection**

---

*/\* Associé à la fonction d'interprétation (IV.167) \*/*

**Entrées :** deux objets  $tst1$  et  $tst2$  de type  $TST$

**Sorties :** un objet  $alist$  de type  $Alist$  qui va contenir les activités appartenant à  $tst1$  et non à  $tst2$

vérifier la validité de  $tst1$  et  $tst2$ , si c'est le cas :

$alist := \emptyset;$

**si**  $card(tst1) \neq 0$  et  $card(tst2) \neq 0$  **alors**

**pour** chaque  $a \in tst1.A$  **faire**

**si**  $a \in tst2.A$  **alors**

$alist := alist \cup \{a\};$

**fin si**

**fin pour**

**fin si**

**retourner**  $alist$ ;

---

---

**Algorithme 17 Difference**

---

/\* Associé à la fonction d'interprétation (IV.169) \*/

**Entrées** : deux objets  $tst1$  et  $tst2$  de type  $TST$

**Sorties** : un objet  $res$  de type  $TST$  représentant le premier  $tst$  résultat de la différence entre  $tst1$  et  $tst2$

vérifier la validité de  $tst1$  et  $tst2$ , si c'est le cas :

$res := Make()$ ;

parcourir l'ensemble des activités de  $tst1$  et de  $tst2$  et retrouver la position  $j$  dans  $tst1$  de la première activité qui appartient à  $tst1$  et n'appartient pas à  $tst2$

**si**  $j \neq 0$  **alors**

**tant que**  $j \leq card(tst1)$  **et**  $tst1.A.a_j \notin tst2.A$  **faire**

$res := Add(res, tst1.A.a_j, tst1.D.d_j)$ ;

$j := j + 1$ ;

**fin tant que**

**fin si**

**retourner**  $res$ ;

---

---

**Algorithme 18 Activities\_Difference**

---

/\* Associé à la fonction d'interprétation (IV.171) \*/

**Entrées** : deux objets  $tst1$  et  $tst2$  de type  $TST$

**Sorties** : un objet  $alist$  de type  $Alist$  qui va contenir les activités appartenant à  $tst1$  et non à  $tst2$

vérifier la validité de  $tst1$  et  $tst2$ , si c'est le cas :

$alist := \emptyset$ ;

**si**  $card(tst1) \neq 0$  **et**  $card(tst2) \neq 0$  **alors**

**pour** chaque  $a \in tst1.A$  **faire**

**si**  $a \notin tst2.A$  **alors**

$alist := alist \cup \{a\}$ ;

**fin si**

**fin pour**

**sinon**

$alist := tst1$ ;

**fin si**

**retourner**  $alist$

---

---

---

# Annexe B

---

## Données Benchmark

### 1 L'enquête Origine-Destination

La base de données que nous avons utilisé est, en fait, une base de déplacements de personnes qui ont répondu à une enquête O-D conduite en 2001 par la Société de Transport de la Communauté Urbaine de Québec (STCUQ). Chaque ligne de cette base contient des informations caractérisant le déplacement d'une personne. Pour plus de clarté dans la description, nous les classons en quatre catégories :

- **Des données attributaires** : ce sont des données qui caractérisent la personne qui a effectué le déplacement ou le ménage auquel appartient la personne en question. Une sélection des données attributaires de la base O-D, qui nous intéressent, est présentée dans le tableau [B.1](#).
- **Des données spatiales** : ce sont des données qui caractérisent géographiquement ou géométriquement un déplacement ou une personne. Les données à caractère spatiale de la base O-D, que nous estimons les plus pertinentes, sont récapitulées dans le tableau [B.2](#).
- **Des données temporelles** : ce sont des données qui caractérisent temporellement un déplacement. Les attributs de la base O-D caractérisant temporellement un déplacement sont présentées par le tableau [B.3](#).
- **Des données sémantiques** : ce sont des données qui décrivent la raison d'un déplacement et les modes de transport utilisés pour effectuer ce déplacement voir tableau [B.4](#).

### 2 Pré-traitement des données Benchmark

Dans cette partie, nous faisons une association entre les entités qui composent notre type TST et qui figurent dans le modèle de données que nous utiliserons lors de l'expérimentation et entre les données benchmark que nous avons en notre possession. À partir de ce travail, nous

Nom	Type	Description
Uniquedepl	Decimal	identifiant unique du déplacement
Uniquepers	Decimal	identifiant unique de la personne qui a effectué le déplacement
Sexe	Integer	genre de la personne qui a effectué le déplacement
Age	Integer	âge de la personne qui a effectué le déplacement
Permiscond	Integer	indique si la personne possède un permis de conduire
Lpasserbus	Integer	indique si la personne possède un laissez-passer d'autobus
Utilautotr	Integer	indique si la personne utilise habituellement l'automobile pour aller à son principal lieu d'activité
Uniquemena	Decimal	identifiant unique du ménage
Nbauto	Integer	nombre d'automobiles dans le ménage
Nbpers	Integer	nombre de personnes dans le ménage

Tableau B.1 – Données attributaires de la base de données benchmark

Nom	Type	Description
Noeudori	Integer	identifiant du noeud d'origine, du déplacement, sur le réseau routier de la région d'étude
Noeuddes	Integer	identifiant du noeud destination, du déplacement, sur le réseau routier de la région d'étude
Origlon	Decimal	longitude du lieu d'origine du déplacement
Origlat	Decimal	latitude du lieu d'origine du déplacement
Destlon	Decimal	longitude du lieu de destination du déplacement
Destlat	Decimal	latitude du lieu de destination du déplacement
Residlon	Decimal	longitude du lieu de résidence de la personne qui a effectué le déplacement
Residlat	Decimal	latitude du lieu de résidence de la personne qui a effectué le déplacement
Occuplon	Decimal	longitude du lieu d'activité principal de la personne qui a effectué le déplacement
Occuplat	Decimal	latitude du lieu d'activité principal de la personne qui a effectué le déplacement

Tableau B.2 – Données spatiales de la base de données benchmark

Nom	Type	Description
Heuredepar	Integer	heure de départ du déplacement
Dateenquet	Date	date du déplacement
Tempauto	Decimal	Durée en automobile estimée du trajet

Tableau B.3 – Données temporelles de la base de données benchmark

Nom	Type	Description
Déplmotif	Integer	motif du déplacement
Mode1	Integer	premier mode de transport utilisé pour effectuer le déplacement
Mode2	Integer	second mode de transport utilisé pour effectuer le déplacement
Mode3	Integer	troisième mode de transport utilisé pour effectuer le déplacement

**Tableau B.4** – Données sémantiques de la base de données benchmark

dégageons les données benchmark qui seront utilisées pour la création de la base expérimentale. Certaines de ces données seront utilisées telles quelles alors que d'autres devront être modifiées ou déduites. Nous décrirons par la suite, les étapes de modifications ou de création de ces dernières.

## 2.1 Analogie entre données benchmark initiales et données expérimentales

Afin d'identifier les traitements nécessaires sur les données benchmark initiales, nous établissons une analogie entre ces dernières et les données nécessaires pour créer notre base de données expérimentale. Cette analogie est exposée dans le tableau B.5.

D'après ce tableau analogique, nous pouvons remarquer que certaines données benchmark vont être utilisées sans subir de modifications, comme les données décrivant les municipalités, les routes, les ménages et certains attributs d'une personne (comme son identifiant ou son sexe). Par contre, certaines données doivent être construites à partir de données existantes. À titre d'exemple, la trajectoire spatio-temporelle d'une personne durant une journée bien déterminée (de type TST) doit être construite à partir de données benchmark ou de données générées (voir B.6).

En effet, comme nous l'avons déjà mentionné, un objet de type TST est composé par un ensemble d'activités,  $a$ , et de déplacements,  $d$ . Le tableau B.6 montre qu'il est nécessaire d'effectuer les traitements suivants sur les données benchmark pour pouvoir créer l'attribut  $Traj$  de type TST de la base de données expérimentale :

- changer le type des temps de départ des déplacements de *Integer* vers *timestamp*,
- construire certains attributs tels que : les temps d'arrivée des déplacements ( $Traj.t_e$ ) ou les lieux d'origine et de destination des déplacements qui doivent être de type *geomtry* (plus précisément des *Point*).

Nous exposerons dans ce qui suit, les différentes étapes de construction de ces données.

Données finales		données initiales	
Nom	Type	Nom	type
Menage.Id	integer	Uniquemena	Decimal
Menage.NbrePersonnes	integer	Nbpers	Integer
Personne.Id	integer	Uniquepers	Decimal
Personne.traj	TST	-	-
Personne.Journée	date	Dateenquet	Date
Personne.Sexe	character	Sexe	Integer
Motif.Id	Integer	-	-
Motif.Nom	charaxter varying	Deplmotif	Integer
Mode.Id	Integer	-	-
Mode.Nom	character varying	Mode1, Mode2 et Mode3	Integer
Lieu.Id	Integer	-	-
Lieu.Localisation	geometry	-	-
Lieu.Description	character varying	-	-
Route.Id	Integer	Reseau.Id	Decimal
Route.Line	geometry	Reseau.the_geom	geometry
Route.Description	character varying	Reseau.Descriptio	character varying
Municipalité.Id	Integer	Municipalité.Id	Integer
Municipalité.zone	geometry	Municipalité.zone	geometry

**Tableau B.5** – Analogie entre données expérimentales finales et données benchmark initiales nécessaires pour la création de la B.D expérimentale

Données expérimentales finales		Données benchmark initiales	
Nom	Type	Nom	type
<i>traj.d.ts</i>	timestamp	Heuredepar	Integer
<i>traj.d.te</i>	timestamp	-	-
<i>traj.d.l<sub>s</sub></i>	geometry	Origlon et Origlat	Decimal
<i>traj.d.l<sub>e</sub></i>	geometry	Destlon et Destlat	Decimal
<i>traj.d.path</i>	geometry	-	-
<i>traj.d.mode</i>		Mode1, Mode2 et Mode3	Integer
<i>traj.d.motif</i>	character varying	Deplmotif	Integer
<i>traj.a.ts</i>	timestamp	-	-
<i>traj.a.te</i>	timestamp	-	-
<i>traj.a.l</i>	geomtry	-	-
<i>traj.a.type</i>	character varying	Deplmotif	Integer

**Tableau B.6** – Analogie entre données expérimentales finales et données benchmark initiales nécessaires pour la création de la donnée de type TST

## 2.2 Chargement de la base O-D

Pour pouvoir être utilisées, les données benchmark doivent être chargées dans notre environnement de travail qui est le SGBD spatial PostgreSQL-PostGIS. L'exportation des données benchmark vers PostgreSQL se fait à l'aide de la commande *ogr2ogr*.

## 2.3 Traitement des données temporelles

Pour la création de l'attribut *Traj* de type *TST*, il nous faut les données temporelles suivantes :

1. *Traj.d.t<sub>s</sub>* : Temps de départ d'un déplacement. Cet attribut existe dans la B.D benchmark sous le nom *Heuredepar* qui est de type *Integer*. Nous avons, juste, eu à modifier son type vers *timestamp*.
2. *Traj.d.t<sub>e</sub>* : temps d'arrivée d'un déplacement. Cette information n'existe pas dans l'enquête O-D qui nous a été fournie. Idéalement et pour que notre expérimentation soit la plus proche de la réalité, nous aurions besoin de cette information ou de la durée 'réelle' des déplacements effectués. Alors que, nous ne possédons pas ces informations et nous n'avons que la donnée *Tempauto* (voir B.3) qui est la durée estimée des trajets effectués en automobile. Nous avons donc choisi d'utiliser cette information pour calculer le temps d'arrivée des déplacements ( $Traj.d.t_e = Traj.d.t_s + Tempauto$ ), bien qu'elle puisse donner des résultats "incohérents". Nous effectuons, par la suite un "nettoyage" des résultats en éliminant les résultats incohérents, c'est-à-dire les déplacements ayant des temps d'arrivée dépassant les temps de départ du déplacement suivant.
3. *Traj.a.t<sub>s</sub>* : Temps de départ d'une activité. D'après la spécification de notre modèle, cette donnée coïncide avec le temps d'arrivée du déplacement qui la précède auquel on rajoute une unité de temps, qui est ici l'unité seconde. Il nous a suffi, donc, juste de se conformer à cette spécification pour alimenter notre B.D.
4. *Traj.a.t<sub>e</sub>* : Temps d'arrivée d'une activité. D'après la spécification de notre modèle, cette donnée coïncide avec le temps de départ du déplacement qui la suit auquel on retranche une unité de temps, c'est à dire on retranche une seconde. Il nous a suffi, donc, juste de se conformer à cette spécification pour alimenter notre B.D. Vu que l'enquête O-D ne mentionne que les horaires de départ des déplacements. L'estimation du temps de fin de la dernière activité du TST pose problème. Nous avons donc émis la supposition que la dernière activité dure une minute pour toute personne.



## 2.4 Traitement des données spatiales

Pour la création de l'attribut *Traj* de type *TST*, il nous faut les données spatiales suivantes :

1. *Traj.d.l<sub>s</sub>* et *Traj.d.l<sub>e</sub>* : respectivement les origines et destinations spatiales d'un déplacement. Ces données doivent être de type *Point*. Nous construisons, donc les points leur correspondant à partir des longitudes et latitudes existantes dans l'enquête O-D (voir **Annex**).
2. *Traj.d.path* : Trajet parcouru durant le déplacement. Dans l'enquête O-D, cette information n'existe pas. Nous avons donc émis la supposition que le trajet parcouru est celui du plus court chemin entre l'origine et la destination du déplacement. Nous avons créé cet attribut en générant des trajets spatiaux de type *geomtry* et ceci à l'aide du module *pgrouting* intégré à Postgis. Pour ce faire, la fonction *shortest\_Path*, appliquée au réseau urbain et à l'enquête O-D a été utilisée.
3. *Traj.a.l* : D'après notre modèle, le lieu d'activité d'une activité correspond au lieu de destination du déplacement qui la suit. Nous alimentons donc notre B.D, et plus particulièrement l'attribut *Traj*, en se conformant à cette spécification.

## 2.5 Traitement des données sémantiques

Les données sémantiques nécessaires à l'alimentation de notre base expérimentale sont les suivantes :

1. *Traj.d.mode* : Modes de transport utilisés pour l'actuel déplacement. Cette composante doit être de type tableau de caractères. L'enquête O-D mentionne pour chaque déplacement les trois modes utilisés, sous la forme d'entiers. Il nous a suffi de décoder ces entiers en chaînes de caractère et de créer pour chaque TST un tableau de chaînes de caractère mentionnant les modes utilisés.
2. *Traj.d.motif* : Motif du déplacement. Cette information existe dans l'enquête O-D mais elle est codée sous la forme d'entiers. Il suffit de décoder l'information et de convertir le type en chaîne de caractère.
3. *Traj.a.type* : Nature de l'activité courante. La nature de l'activité correspond dans notre modèle au motif du déplacement précédant.
4. Les tables *Mode* et *Motif* : ces tables sont des tables de tests que nous voulons créer pour pouvoir émettre des requêtes sur les différents modes et motifs de déplacements existants. Nous les avons créées, "manuellement" à partir des différents modes et motifs

de déplacements existants dans l'enquête O-D. Nous nous sommes appuyées pour cela sur les fichiers de codage de l'enquête O-D.

---

## Bibliographie

- [1] *Spatial and Temporal Reasoning in Geographic Information Systems*. 1998. [92](#)
- [2] Fank A., Rapper J., and Cheylan J.-P. Life and motion of socio-economic units. *New York, Londres : Taylor and Francis, GISDATA Series*, 2001. [13](#)
- [3] Frihida A., Zheni D., Ben Ghezala H., and Claramunt C. Modeling trajectories : A spatio-temporal data type approach. In *20th International Workshop on Database and Expert Systems Application*. [135](#)
- [4] Frihida A., Marceau D.J., and Thériault M. Dimension temporelle et modélisation d’une animation cartographique dans un sig orienté objet. *Revue Internationale de Géomatique*, 13(1) :107–127, 2003. [177](#)
- [5] Frihida A., Marceau D.J., and Thériault M. Development of a temporal extension to query travel behavior time paths using an object-oriented gis. *geoIn*, 8(3) :211–235, 2004. [52](#), [80](#)
- [6] Frihida A., Marceau D. J., and Thériault M. Spatio-temporal object-oriented data model for disaggregated travel behaviour. *tgis*, 6(3) :277–294, 2002. [2](#), [52](#), [58](#), [169](#)
- [7] Sistla A., Wolfson O., Chamberlain S., and Dao S. Modeling and querying moving objects. In *Proceedings IEEE International Conference on Data Engineering (ICDE)*, pages 422–433, 1997. [35](#)
- [8] Roshannejad A.A and Kainz W. Handling identities in spatio-temporal databases. *Auto-Carto 1, Charlotte, North Carolina*, 1995. [12](#)

- 
- [9] Lenntrop Bo. Path in space-time environments : A time-geographic study of the movement possibilities of individuals. *Lund Studies in Geography*, 44 :155, 1976. [2](#), [28](#)
- [10] Du Mouza C. and Rigaux P. *Bases de Données Spatio-Temporelles*, chapter Le temps, l'Espace et l'Evolutif en Science du Traitement de l'Information. 2000. [15](#), [16](#)
- [11] Du Mouza C. and Rigaux P. Mobility patterns. *geoIn*, 9(4) :297–319, 2005. [46](#)
- [12] Lema J. A. C., Forlizzi L., Güting R. H., Nardelli E., and Schneider M. Algorithms for moving objects databases. *cj*, 46(6) :680–712, 2003. [38](#)
- [13] Parent C., Spaccapietra S., Renso C., Andrienko G.L, Andrienko N.V, Bogorny V., Damiani M. L., Gkoulalas-Divanis A., Fernandes de Macêdo J. A., Pelekis N., Theodoridis Y., and Yan Z. Semantic trajectories modeling and analysis. *ACM Comput. Surv*, 2013. [20](#), [43](#), [48](#), [57](#)
- [14] Pfoser D. and Theodoridis Y. Generating semantics-based trajectories of moving objects. In *International Workshop on Emerging Technologies for Geo-Based Applications*, pages 59–76, 2000. [41](#)
- [15] Wang D. and Cheng T. A spatio-temporal data model for activity-based transport demand modelling. *ijgis*, 15(6) :561– 585, 2001. [2](#), [34](#), [51](#), [57](#)
- [16] Zheni D., Frihida A., and Ben Ghézala Hajjami H. A semantic approach for the modeling of trajectories in space and time. In *ER Workshops 2009, LNCS 5833*. [152](#)
- [17] Zheni Triki D., Frihida A., Ben Ghezala H., and Claramunt C. Modèle et langage de manipulation de trajectoires spatio-temporelles. *Revue Internationale de Géomatique*, 2010. [152](#)
- [18] Sinton D.F. The inherent structure of information as a constraint to analysis : Mapped thematic data as a case study. *Harvard Papers on GIS*, 1978. [12](#)
- [19] Site du ministère des transports du Quebec. <http://www.mtq.gouv.qc.ca>. Dernière consultation en juin 2014. [154](#)
- [20] Bouhali E., Turki S. Y., and Ben Medien O. Politique de stationnement dans les centres villes : Enjeux et effets sur les usages de l'espace. le cas du centre ville de tunis. In *AGOUMY T. et REFASS M., Villes et environnement durable en Afrique et au Moyen Orient*. [151](#)

## Bibliographie

---

- [21] Clementini E., Di Felice P., and Van Oosterom P. Formalization of topological knowledge in spatial query languages. In *SEBD*, pages 57–71, 1993. [92](#), [122](#)
- [22] Documentation en ligne de PostgreSQL. <http://docs.postgresqlfr.org/9.3/>. Dernière consultation en juin 2014. [144](#), [156](#)
- [23] Allen J. F. Towards a general theory of action and time. *ai*, 23(2) :123–154, 1984. [23](#), [53](#), [65](#), [77](#), [97](#), [122](#), [129](#)
- [24] Schreiber F.A. Is time a real time? an overview of time ontology in informatics. In *In Real Time Computing*, pages 283–307. SpringerVerlag, 1994. [22](#)
- [25] P. Forer, Chen H. F., , and J. F. Zhao. Building, unpacking and visualising human flows with gis. In *Proceedings of the GIS research UK 12th Annual Conference*, pages 334–336, University of East Anglia, 2004. [19](#)
- [26] Cattel R. G. G. and Barry D. K. *The Object Database Standard : ODMG 2.0*. Morgan Kaufmann Publishers, 1997. [38](#)
- [27] Güting R. H., Almeida V. T., and Ding Z. Modeling and querying moving objects in networks. *vldbj*, 15(2) :165–190, 2006. [40](#), [42](#), [56](#)
- [28] Yu H. and Shaw S.-L. Representing and visualizing travel diary data : A spatio-temporal gis approach. In *ESRI International User Conference*. [49](#), [57](#)
- [29] Miller H.J. A measurement theory for time geography. *Geographical Analysis*, 2005. [26](#), [29](#), [43](#), [48](#), [54](#), [57](#), [169](#)
- [30] Kakoudakis I. *The TAU Temporal Object Model*. PhD thesis, University of Manchester Institute of Science and Technology, 1996. [23](#), [38](#)
- [31] Sandu-Popa I., Kharrat A., and Zeitouni K. Base de données de capteurs à localisation mobile modèle et langage. *Systèmes d'Information et géolocalisation, Revue RSTI Série ISI (Ingénierie des Systèmes d'Information)*, Editions Hermès, 14(5) :35–58, 2009. [40](#), [56](#)
- [32] Sandu Popa I. and Zeintouni K. Managing sensor data with mobile location. In *International Conference on Spatial Analysis and GEomatics, Research & Developments (SAGEO 2008)*, Juin 2008. [40](#)
- [33] Guttag J. Abstract data types and the development of data structures. *Communications of the ACM*, 20(6) :297–404, 1977. [62](#)

- 
- [34] Ma J. and Knight B. A general temporal theory. *cj*, 37(2) :114–123, 1994. [97](#)
- [35] Miller H. J. Modeling accessibility using space-time prism concepts within geographical information systems. *ijgis*, 5(3) :287–301, 1991. [2](#)
- [36] Miller H. J. Activities in space and time. In *Handbook of Transportation Research 5 : Transport Geography and Spatial Systems*, Pergamon/Elsevier Science, pages 647–660. Pergamon/Elsevier Science, 2004. [29](#), [43](#)
- [37] Xu J. and Güting R.H. A generic data model for moving objects. *geoIn*, 2013. [44](#), [53](#), [54](#), [57](#), [169](#)
- [38] Follin J.M, Moreau G., Bouju A., Polidori L., and Soussi M. Enrichissement d’une base de données routière à partir de trajectoires gps de véhicules d’urgence. application à l’aide au choix d’itinéraires. *Ingénierie des Systèmes d’Information*, 2012. [42](#), [53](#), [56](#)
- [39] Forlizzi L., Güting R. H., Nardelli E., and Schneider M. A data model and data structures for moving objects databases. In *Proceedings ACM SIGMOD Intl. Conf. on Management of Data*, pages 319–330, 2000. [37](#), [38](#)
- [40] G. Langran. *Time in Geographic Information Systems*. London, Taylor & Francis, 1992. [12](#)
- [41] Egenhofer M. Definitions of line line relations for geographic databases. *Data Engineering*, ? [21](#)
- [42] Egenhofer M. and Mark D. Modelling conceptual neighborhoods of topological line-region relations. *International Journal of Geographical Information Systems*, 1995. [21](#)
- [43] Egenhofer M. and Franzosa R.D. Point set topological relations. *International Journal of Geographical Information Systems*, 1991. [21](#)
- [44] Erwig M., Güting R. H., Schneider M., and Vazirgiannis M. Abstract and discrete modeling of spatio-temporal data types. In *6th ACM Symposium on Geographic Information Systems*, pages 131–136, 1998. [36](#)
- [45] Erwig M. and Schneider M. Spatio-temporal predicates. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 14(4) :881–901, 2002. [37](#)
- [46] Erwig M. and Schneider M. Stql-a spatio-temporal query language. *Mining Spatio-Temporal Information Systems*, 669 :105–126, 2002. [38](#)

## Bibliographie

---

- [47] Mark D. M., Egenhofer M. J., Hornsby K. Bian L., Rogerson P., and Vena J. Spatio-temporal gis analysis for environmental health using geospatial lifelines. In *A. Flahault, L. Toubiana, and A. Valleron (Eds), 2nd International Workshop on Geography and Medicine, GEOMED '99*, page 52, Paris, France, 1999. [26](#), [28](#)  
iault02
- [] Thériault M., Claramunt C., Seguin A. M., and Villeneuve P. Temporal gis and statistical modelling of personal lifelines. In *9th Spatial Data Handling symposium*, pages 433–450. D. Richardson and P. van Oosterom (eds.), Springer-Verlag, 9-12 Juillet 2002. [2](#)  
iault99
- [] Thériault M., Claramunt C., and Villeneuve P.Y. A spatio-temporal taxonomy for the representation of spatial set behaviours. *Spatio-temporal database management*, 1999. [14](#)
- [50] Yuan M. Modeling semantical, temporal, and spatial information in geographic information systems. *M. Craglia and H. Couclelis eds. Geographic Information Research : Bridging the Atlantic, Taylor & Francis*, 1996. [12](#)
- [51] Sakr M.A. and Güting R.H. A new approach for spatiotemporal pattern queries in trajectory databases. In *11th Intl. Conference on Mobile Data Management*. [38](#), [54](#), [56](#)
- [52] Andrienko N., Andrienko G., Pelekis N., and Spaccapietra S. *Basic concepts of movement data*, chapter Mobility, Data Mining and Privacy, Geographic Knowledge Discovery. Springer, 2008. [19](#)
- [53] Pelekis N. and Theodoulidis B. Tau tll data cartridge. Technical report, Rapport Technique TR-02-3,CRIM, UMIST, Manchester, Grande-Bretagne, 2002. [38](#)
- [54] Pelekis N., Theodoulidis B., Kopanakis I., and Theodoridis Y. Literature review of spatio-temporal database models. *The Knowledge Engineering Review Journal*, 19(3) :235–274, 2005. [14](#), [38](#), [53](#), [78](#)
- [55] Pelekis N., Frentzos E., Giatrakos N., and Theodoridis Y. Hermes : A trajectory db engine for mobility-centric applications. *International Journal of Knowledge-based Organizations*, 20014. [38](#)
- [56] Pelekis N., Frentzos E., Giatrakos N., and Theodoridis Y. Hermes : aggregative lbs via a trajectory db engine. In *Proceedings of the 2008 ACM SIGMOD international conference*

- on Management of data*, SIGMOD '08, pages 1255–1258, New York, NY, USA, 2008. ACM. [38](#)
- [57] Pelekis N., Theodoridis Y., Vosinakis S., and Panayiotopoulos T. Hermes - a framework for location-based data management. In *the Proceedings of the 10th International Conference on Extending Database Technology (EDBT06)*, pages 1130–1134. LNCS 3896, 2006. [3](#), [38](#), [56](#), [170](#)
- [58] Alvares L. O., Oliveira G., Heuser C. A., and Bogorny V. A framework for trajectory preprocessing for data mining. In *Proc. of the 21st International Symposium on Software Engineering and Knowledge Engineering, (SEKE'09)*, 1-3 July 2009. [33](#), [48](#)
- [59] Alvares L. O., Bogorny V., Kuijpers B., de Macedo J. A. F., Moelans B., and Vaisman A. A model for enriching trajectories with semantic geographical information. In *Proceedings of the ACM 15th International Symposium on Advances in Geographic Information Systems (ACM-GIS'07)*, pages 162–169, 7-9 Novembre 2007. [48](#), [57](#)
- [60] Klein O. *Modélisation et représentations spatio-temporelles des déplacements quotidiens urbains : Application à l'Aire Urbaine Belfort-Montbéliard*. PhD thesis, Université Louis Pasteur Strasbourg, Juin 2007. [27](#)
- [61] Wolfson O., Xu B., Chamberlain S., and Jiang L. Moving objects databases : Issues and solutions. In *Proceedings of the 10th Int. Conference on Scientific and Statistical Database Management*, pages 111–122, 1998. [2](#), [35](#), [56](#)
- [62] Wolfson O., Sistla P., Chamberlain S., and Yesha Y. Updating and querying databases that track mobile units. *Special issue of the Distributed and Parallel Databases Journal (DAPD) on Mobile Data Management and Applications*, 7(3) :257–387, 1999. [3](#), [33](#), [35](#), [53](#)
- [63] Wolfson O., NChamberlain S., Sistla P., Xu B., and Zhou J. Domino : Databases for moving objects tracking. In *Proceedings of the ACM-SIGMOD 1999, International Conference on Management of Data*, pages 547–549, 1999. [35](#)
- [64] Site officiel du Open Geospatial Consortium. <http://www.opengeospatial.org/>. [144](#)
- [65] Site officiel du système d'information géographique QGIS. <http://www.qgis.org/fr/site/>. Dernière consultation en juin 2014. [157](#)



## Bibliographie

---

- [66] Jin P., Wan S., and Yue L. Conceptual modeling for moving object applications. In *The 9th International Conference on Mobile Data Management (MDM'08)*, pages 217 – 218. IEEE CS Press, 2008. [42](#), [56](#)
- [67] Laube P. *Analysing Point Motion ?Spatio-Temporal Data Mining of Geospatial Lifelines*. PhD thesis, Université de Zürich, 2005. [15](#), [16](#)
- [68] Abler R., Adams J. S., and Gould P. Spatial organization : The geographer's view of the world. *NJ : Prentice-Hall*, 1971. [12](#)
- [69] Praing R. and Schneider M. A universal abstract model for future movements of moving objects. In *Proc of the 10th AGILE Int. Conf. on Geographic Information Science*, pages 11–120. Springer, 2007. [38](#), [54](#), [56](#)
- [70] Sondgrass R. and Ahn I. A taxonomy of time in databases. In *Proceedings of SIGMOD '85 Proceedings of the 1985 ACM SIGMOD International Conference on Management of Data*, 1985. [23](#)
- [71] Güting R.H., Böhlen H., Erwig M., Jensen C. S., and Lorentzos N. A.and Schneider M.and Vazirgiannis M. A foundation for representing and quering moving objects. *tdbs*, 25(1) :1–42, 2000. [2](#), [3](#), [33](#), [36](#), [37](#), [38](#), [40](#), [41](#), [44](#), [53](#), [56](#), [78](#), [89](#), [94](#), [175](#)
- [72] Güting R.H., Behr T., , and Düntgen C. Secondo : A platform for moving objects database research and for publishing and integrating research implementations. *IEEE Data Engineering Bulletin*, 33(2) :56–63, 2010. [38](#)
- [73] Brakatsoulas S., Pfoser D., and Tryfona N. Modeling, storing and mining moving object databases. In *Proc. IDEAS conference*. [41](#), [53](#), [56](#)
- [74] Chardonnel S., Du Mouza C., Fauvet M. C., Josselin D., and Rigaux P. Patrons de mobilité : proposition de définition, de méthode de représentation et d'interrogation. In *Colloque Cassini'04 - 7ème conférence du GDR Sigma "Géomatique et Analyse Spatiale"*, pages 19–23, 2-4 Juin 2004. [27](#), [45](#), [57](#)
- [75] Dieker S. and Güting R. H. Plug and play with query algebras : Secondo-a generic dbms development environment. In *Proceedings of the International Database Engineering and Applications Symposium (IDEAS 2000)*, pages 380–390, Septembre 2000. [38](#), [45](#)

- 
- [76] Spaccapietra S., Parent C., Damiani M. L., De Macedo J.A. and Porto F., and Vangenot C. A conceptual view on trajectories. *Data & Knowledge Engineering*, 65(1) :126–146., 2008. [17](#), [25](#), [26](#), [43](#), [47](#), [48](#), [57](#), [169](#)
- [77] Joseph E. Stoy. *Denotational Semantics : The Scott-Strachey Approach to Programming Language Theory*. MIT Press, Cambridge, MA, USA, 1977. [62](#)
- [78] Hägerstrand T. What about people in regional science? *Papers of the Regional Science Association*, 1970. [2](#), [4](#), [10](#), [27](#), [33](#)
- [79] Bogorny V., Kuijpers B., , and Alvares L. O. St-dmql : A semantic trajectory data mining query language. *IJGIS International Journal of Geographical Information Science*, 23(10) :1245–1276, 2009. [48](#)
- [80] Bogorny V., Renso C., Ribeiro A., Siqueira F. L., and Alvares L. O. Constant : A conceptual data model for semantic trajectories of moving objects. *Transactions in GIS*, 2014. [48](#), [57](#)
- [81] Bogorny V., Avancini H., Cesar de Paula B., Kuplich C. R., and Alvares L. O. Weka-stpm : a software architecture and prototype for semantic trajectory data mining. *Transactions in GIS*, 15(2) :227–248, 2011. [48](#)
- [82] Noyon V. PhD thesis, Ecole Nationale Supérieure d’Arts et Métiers. [36](#)
- [83] Noyon V., Claramunt C., and Devogele T. A relative representation of trajectories in geographical spaces. *geoIn*, 4(11) :479–496, 2007. [21](#), [35](#), [56](#)
- [84] Noyon V., Devogele T., and Claramunt C. A formal model for representing point trajectories in two-dimensional spaces. In *Proceedings of the Conference on Conceptual Modelling and GIS (COMOGIS’05)*, pages 208–217. Lecture Notes in Computer Science, Springer, Octobre 2005. [35](#)
- [85] M.F. Worboys. Unifying the spatial and temporal components of geographical information. In *Advances in Geographic Information Systems : Proceedings of the International Symposium on Spatial Data Handling*, pages 505–517. Healey, R. and Waugh, T. (eds.), Taylor and Francis, 1994. [24](#)
- [86] Li X., Claramunt C., Ray C., and Lin H. A semantic-based approach to the representation of network-constrained trajectory data. In *Proceedings of the 12th International Symposium on Spatial Data Handling (SDH 2006)*, pages 451–464. Springer-Verlag, Juillet 2006. [33](#), [34](#), [39](#), [56](#)

## Bibliographie

---

- [87] Yan Z., Chakraborty C., Spaccapietra S., and Aberer K. Semitri : A framework for semantic annotation of heterogeneous trajectories. In *Fourteenth International Conference on Extending Database Technology (EDBT 2011)*. [48](#)