Spatio-Temporal Modeling of Urban Road Traffic

Kamaldeep Singh Oberoi

Normandie Université

THÈSE

Pour obtenir le diplôme de Doctorat

Spécialité Informatique

Préparée au sein de l'Université de Rouen Normandie, ED MIIS

Modélisation Spatio-Temporelle du Trafic Routier en Milieu Urbain

Présentée et soutenue par

Kamaldeep Singh OBEROI

Thèse soutenue publiquement le Lundi 18 November 2019
devant le jury composé de

Dr. Arnaud CASTEIGTS	Maître de conférences, LaBRI, Université de Bordeaux, France	Rapporteur
Dr. Samira EL YACOUBI	Professeure, Espace-Dev, Université de Perpignan, France	Rapporteure
Dr. John STELL	Senior Lecturer, School of Computing, University of Leeds, U.K.	Examinateur
Dr. Jérôme GENSEL	Professeur, LIG, Université Grenoble Alpes, France	Examinateur
Dr. Géraldine DEL MONDO	Maître de conférences, LITIS, INSA Rouen, France	Encadrante
Dr. Yohan DUPUIS	Responsable Ingénierie, IRSEEM, ESIGELEC, France	Encadrant
Dr. Pascal VASSEUR	Professeur, LITIS, Université de Rouen, France	Directeur de thèse

UNIVERSITÉ DE ROUEN NORMANDIE

MI∫IΣ

litis

SPATIO-TEMPORAL MODELING OF URBAN ROAD TRAFFIC

Kamaldeep Singh OBEROI

Université de Rouen Normandie, France
18 November 2019

Résumé

Le domaine de la modélisation du trafic routier vise à comprendre son évolution. Dans les dernières années, plusieurs modèles du trafic ont été proposés dans l'objectif de géolocaliser embouteillages au sein du trafic, détecter des motifs dans le trafic routier, estimer l'état du trafic etc. La plupart des modèles proposés considèrent le trafic routier en terme de ses constituants ou comme une entité agrégée en fonction de l'échelle choisie et expliquent l'évolution du trafic quantitativement en tenant compte des relations entre les variables de trafic comme *le flot*, *la densité* et *la vitesse*. Ces modèles décrivent le trafic en utilisant des données très précises acquises par différents capteurs. La précision des données rend son calcul coûteux en termes de ressources requises. Une des solutions à ce problème est la représentation qualitative du trafic routier qui réduit le nombre de ressources de traitement nécessaires.

Puisque le trafic routier est un phénomène spatio-temporel, les modèles proposés pour représenter ce type de phénomène pourraient être appliqués dans le cas du trafic routier. Les modèles spatio-temporels, proposés par la communauté de l'Analyse Spatio-Temporelle, ont comme objectif la représentation d'un phénomène tant du point de vue qualitatif que quantitatif. Certains de ces modèles proposent une discrétisation des phénomènes modélisés en considérant un phénomène comme constitué d'entités. Appliquée au trafic routier, cette notion permet d'identifier différents entités, comme les véhicules, les piétons, les bâtiments etc., qui le constituent. Ces entités influent sur l'évolution du trafic.

Les modèles spatio-temporels qualitatifs définissent l'effet des différentes entités les unes sur les autres en terme de relations spatiales. L'evolution spatio-temporelle du phénomène modélisé est représenté par la variation temporelle de ces relations. La prise en compte des entités du trafic et des relations spatiales formalise une structure qui peut être représentée en utilisant un graphe, où les noeuds modélisent des entités et les arcs des relations spatiales. Par conséquent, l'evolution du trafic, modélisée via ce graphe, devient l'évolution du graphe et peut être représenté en terme de la variation de la structure du graphe ainsi que celle des attributs de ses noeuds et de ses arcs. Dans cette thèse, nous proposons une modélisation du trafic routier de ce type basée sur la théorie des graphes.

Une des applications à la modélisation du trafic routier est la détection des motifs pertinents au sein du trafic. Dans les modèles du trafic existants, les motifs détectés sont *statistique* et sont représentés en utilisant des caractéristiques numériques. Le modèle que nous proposons dans cette thèse met en avant la structure représentant le trafic routier et peut donc être utilisé pour définir des motifs *structurels* du trafic qui prennent en compte des différentes entités du trafic et leurs relations. Ces motifs structurels sont sous-jacent à une modélisation sous forme de graphe dynamique. Dans cette thèse, nous proposons un algorithme pour détecter ces motifs structurels du trafic dans le graphe spatio-temporel représentant le trafic routier. Ce problème est formalisé comme celui de *l'isomorphisme de sous-graphe* pour des graphes dynamiques. L'algorithme proposé est evalué en fonction des différents paramètres de graphes.

Mots-clés: Modélisation du trafic routier, Modélisation qualitative, Modélisation spatio-temporelle, Théorie des graphes, Graphes dynamiques, Détection de motifs, Motifs structurels, Isomorphisme de sous-graphe

Abstract

For past several decades, researchers have been interested in understanding traffic evolution, hence, have proposed various traffic models to identify bottleneck locations where traffic congestion occurs, to detect traffic patterns, to predict traffic states etc. Most of the existing models consider traffic as many-particle system, describe it using different scales of representation and explain its evolution quantitatively by deducing relations between traffic variables like *flow*, *density* and *speed*. Such models are mainly focused on computing precise information about traffic using acquired traffic data. However, computation of such precise information requires more processing resources. A way to remedy this problem is to consider traffic evolution in qualitative terms which reduces the required number of processing resources.

Since traffic is spatio-temporal in nature, the models which deal with spatio-temporal phenomenon can be applied in case of traffic. Such models represent spatio-temporal phenomenon from qualitative as well as quantitative standpoints. Depending on the intended application, some models are able to differentiate between various entities taking part in the phenomenon, which proves useful in case of traffic since different objects like vehicles, buildings, pedestrians, bicycles etc., directly affecting traffic evolution, can be included in traffic models.

Qualitative spatio-temporal models consider the effects of different entities on each other in terms of spatial relations between them and spatio-temporal evolution of the modeled phenomenon is described in terms of variation in such relations over time. Considering different traffic constituents and spatial relations between them leads to the formation of a structure which can be abstracted using graph, whose nodes represent individual constituents and edges represent the corresponding spatial relations. As a result, the evolution of traffic, represented using graph, is described in terms of evolution of the graph itself, i. e. change in graph structure and attributes of nodes and edges, with time. In this thesis, we propose such a graph model to represent traffic.

As mentioned above, one of the applications of existing traffic models is in detecting traffic patterns. However, since such models consider traffic quantitatively, in terms of acquired traffic data, the patterns detected using such models are *statistical* (a term employed by Pattern Recognition researchers) in the sense that they are represented using numerical description. Since graph-based traffic model proposed in this thesis represents the structure of traffic, it can be employed to redefine the meaning of traffic patterns from *statistical* to *structural* (also a term from Pattern Recognition community). Structural traffic patterns include different traffic constituents and their inter-links and are represented using time-varying graphs. An algorithm to detect a given structural traffic pattern in the spatio-temporal graph representing traffic is proposed in this thesis. It formalizes this problem as *subgraph isomorphism* for time-varying graphs. In the end, the performance of the algorithm is tested using various graph parameters.

Keywords: Traffic Modeling, Qualitative modeling, Spatio-Temporal Modeling, Graph Theory, Time-varying graphs, Pattern Detection, Structural Patterns, Subgraph Isomorphism

Publications

The work described in this thesis has appeared in the following publications.

Oberoi, Kamaldeep S et al. (2017a)
"Spatial Modeling of Urban Road Traffic Using Graph Theory." In: *Spatial Analysis and GEOmatics*, pp. 264–277. DOI: 10.5281/zenodo.1006677.

– (2017b)
"Towards a qualitative spatial model for road traffic in urban environment." In: *2017 IEEE 20th International Conference on Intelligent Transportation Systems*. IEEE, pp. 1724–1729. DOI: 10.1109/ITSC.2017.8317644.

Oberoi, Kamaldeep Singh et al. (2018)
"Modeling Road Traffic Takes Time." In: *10th International Conference on Geographic Information Science*. Vol. 114. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 52:1–52:7. ISBN: 978-3-95977-083-5. DOI: 10.4230/LIPIcs.GISCIENCE.2018.52.

Oberoi, Kamaldeep Singh et al. (2019)
"Detecting Structural Patterns In Road Traffic Using Dynamic Subgraph Isomorphism." In: *Pattern Recognition*. In Preparation.

Acknowledgements

The journey from the beginning until the end of PhD is full of ups and downs. At both *minima* and *maxima* of this journey, we need people who could support and encourage us as well as share our joy and happiness. I consider myself incredibly lucky to have met some amazing bunch of people who were there for me during the last three years.

I want to start by thanking my thesis director Dr. Pascal Vasseur for his trust in me and for giving me the opportunity to pursue this thesis. We met each other when I started working with him as a research engineer back in 2016. Even at that time, as during my PhD, he was always supportive to pursue different ideas to complete the tasks at hand. He has always tried to provide positive criticism which greatly helps in having an open-minded communication with him.

Next, I want to thank my co-supervisor Dr. Yohan Dupuis for keeping me on track towards proposing an application oriented traffic model and not getting lost in the theoretical realm. During our monthly meetings, he would ask me intriguing questions about my work which helped me think of various things which never would have entered my mind. His knowledge about Graph Theory and Spatio-Temporal Modeling is commendable given that these are not his primary research fields. I hope to have mentors like him in the future as well.

And of course I have to mention my second co-supervisor Dr. Géraldine Del Mondo for whom I don't have words to express gratitude. Without her support, encouragement and guidance, this thesis would not have been possible. She has always been there for me during all the highs and lows of this PhD. She is the best mentor that anyone could ask for. Other than this research topic, we have a shared interest in becoming private pilots and she has motivated and supported my dream of having a private pilot license for which I am forever thankful. I wish her all the best in her journey to become a pilot and hope that we can fly together in the future (of course I will have to get my license first !).

I would also like to thank Dr. Benoît Gaüzère for his support and guidance on the topics related to subgraph isomorphism. Our discussions were quite interesting and I learnt a lot from him. I've also had the chance to teach a course with him and he has always been helpful should there be any problems.

Speaking of teaching, I've met amazing professors and associate professors with whom I've taught various courses at INSA Rouen. I want to thank Dr. Julien Saunier, Dr. Laurent Vercouter, Dr. Nicolas Delestre and Dr. Abdelaziz Bensrhair for their mentorship and guidance in teaching me the *art of teaching*.

A research laboratory or an enterprise cannot function without its secretaries who work behind the scenes in facilitating the lives of everyone working there. The secretaries Ms. Brigitte Diarra, Ms. Sandra Hague, Ms. Fabienne Bocquet and Ms. Marion Baudesson have worked tirelessly in helping me in every and all the ways possible in solving tiniest as well as biggest administrative problems which I faced during my thesis and I am forever grateful to them.

There are so many people whom I met during the last three years, my friends, colleagues who became friends over time and friends who moved away, to whom I am grateful to have been a part of my happiness and for sharing my sorrow and for helping me become a better version of myself over the last three years. I wish each and every one of them all the best in their future endeavours.

In the end, I want to thank my parents for their unconditional support throughout my entire life. They were heartbroken when I came to France in 2014 for my studies but I hope that they are proud of me today for successful completion of my PhD. I wish to keep making them proud in the future as well.

Onto the next adventure now !!

Kamaldeep Singh OBEROI
Rouen, France
November 2019

Contents

List of Figures

List of Tables

Acronyms

NaSch Nagel-Schreckenberg

LWR Lighthill-Whitham-Richards

ITS Intelligent Transportation Systems

ADAS Advanced Driver Assistance Systems

VANET Vehicular Ad-hoc Network

CSP Constraint Satisfaction Problem

1 Overview

Now considered as a sub-domain of the field of Intelligent Transportation Systems (ITS), Traffic Modeling has been an area of research interest since 1940s. It mainly focuses on describing road traffic using mathematical equations relating traffic variables like *flow*, *density* and *speed* etc. and modeling variations in them, either with space and/or time. Such variables are useful for estimating traffic state at a given location or a given point in time, for predicting future traffic states as well as for detecting traffic patterns.

Existing research in this domain represents traffic at different scales. On the one hand, traffic is considered in terms of individual vehicles (*microscopic models*) while on the other hand, it is regarded as a single aggregated entity (*macroscopic models*). Microscopic models incorporate the effect individual vehicles have on the movement of their neighbouring vehicles, due to which they are more computationally challenging. Macroscopic models ignore such minute effects and explain variations in average traffic variables. Hence, they require less processing resources but also provide coarse information about traffic. Combination of microscopic and macroscopic models, called *mesoscopic/hybrid models*, tend to ease this trade-off between required computation and provided level of detail.

Historically, Traffic Modeling has mainly been an empirical domain of research where, firstly, traffic data is acquired using different kinds of data sources and then theoretical models are developed to explain, more or less, the semantics behind the data and relate traffic variables. Since proposed models are data-centric, they formalize traffic from a quantitative point of view. Such data-centric representation of traffic is necessary in case precise information is required. However if one wants to consider traffic in qualitative terms, so as to be able to perform reasoning about traffic, the precision of the data is not the focus. Rather, the idea is to have coarse information which can be processed using fewer resources, even though the results might be imprecise (up to a certain degree).

Moreover, since existing traffic models are dependent on traffic data, collected while considering vehicles as primary traffic constituent, they tend to ignore other objects, like pedestrians, bicycles, buildings, etc. which also have an effect on traffic. This is more problematic in case of urban traffic than highway traffic since such objects have more pronounced effect in towns and cities.

In this thesis, we will try to answer the following questions:

1. How can qualitative knowledge about traffic be computed from quantitative data? How can it be represented in traffic model?
2. What other types of objects affect urban traffic? How can such an effect be modeled?
3. Can qualitative knowledge and effect of various objects on traffic be represented in a single model?
4. Can such a model also have different representation scales like existing traffic models?

Road traffic is a spatio-temporal phenomenon. It is a *phenomenon* because it can be observed either in terms of individual elements or as a single collective "thing". It is *spatio-temporal* because individual elements of traffic move in space over time. In addition, from quantitative standpoint, traffic data describes traffic state at a given point in space and time, and such data, acquired over some spatio-temporal interval, shows the evolution of traffic over that interval.

The field of Spatio-Temporal Modeling deals with the representation of spatio-temporal phenomenon from both quantitative and qualitative standpoints. Database researchers have developed various data models capable of representing such phenomenon in the database which can be queried to extract required information. The models developed consider the phenomenon to be represented in terms of its constituents, referred to as *entities*, which may have attributes associated to them. Geographical databases have additional functionality to extract spatial knowledge about the phenomenon from the stored data. They consider entities in terms of their geometrical primitives (points, lines, polygons) and compute qualitative spatial relations between them using acquired quantitative data. These spatial relations are useful for qualitative reasoning about the modeled phenomenon. In addition to spatial knowledge, some data models consider time, either as a separate dimension or a data attribute, making them spatio-temporal.

Since road traffic is spatio-temporal in nature, it can be modeled using the notions developed in the field of Spatio-Temporal Modeling. Doing so, the constituents of traffic act as entities in the model, about which data is stored in the database, and qualitative spatial relations between various entities provide a tool for reasoning about traffic in qualitative terms. As mentioned before, in case of urban traffic, vehicles are not the only traffic constituent. In addition to vehicles, pedestrians and bicycles also affect traffic flow. Talking about static entities, buildings, traffic signs and road infrastructure also play an important role in forcing the traffic to flow the way it does.

Incorporating such static and dynamic entities, along with spatial relations between them, in the model describing urban traffic, provides a complete picture about the intricacies of traffic flow, which is not captured by mathematical equations describing traffic. The effect of these entities on traffic becomes explicit when variation in spatial relations between them is modeled over time.

Considering various traffic entities and spatial relations between them forms a structure representing traffic, and graphs provide a natural way for its formalization. The nodes of the graph represent traffic entities and edges the spatial relations between two entities. The spatio-temporal nature of traffic is captured by a spatio-temporal graph whose structure as well as the attributes of its nodes and edges vary with time. Such a spatio-temporal graph model of traffic is the main contribution of this thesis.

Defining such a graph for a given urban environment could prove to be challenging due to its size and density. Hence, we propose to reduce the graph size by zooming-in on a particular road segment or intersection and considering only the traffic entities present on them. The size of the graph can be further reduced by segregating the considered road segment in terms of carriageways and sectors. In the proposed model, we classify nodes according to various classes of objects. Since we do not consider spatial relations between all object classes, the number of edges in the graph is restricted, which keeps its density in check.

Incorporating temporal dimension in the graph provides a way to model traffic evolution in terms of graph evolution. We separate the evolution in graph structure and in attributes of its nodes and edges so as to be able to focus on them individually. The variations in graph structure are considered to be instantaneous whereas variations in node and edges attributes might be instantaneous or occur over some time interval. Since both types of variations occur simultaneously, the proposed graph is *fully dynamic*.

Figure 1.1 shows the global workflow for generating spatio-temporal graph representing traffic. Various modules of the workflow are as follows:

- **Data Sources** module focuses on acquiring raw traffic data using different kinds of sensors. Globally, we differentiate between two types - Perception and Non-Perception. Perception sensors are vision-based sensors like Cameras, LiDARs, RADARs etc. which collect images and/or videos of traffic scenes. Such sensors could be mounted on-board vehicles, on roadside infrastructural elements like traffic signs and buildings or at intersections. Non-Perception sensors acquire traffic data such as flow, density, time headways, GPS location of vehicles as well as vehicles' internal data collected using Controller Area Network (CAN) bus. These sensors could also be on-board vehicles, like GPS sensor, CAN bus etc. or off-board like inductive loop counter which is embedded under the road surface. Since this is a global workflow, we consider many types of sensors. However, it can be restricted according to the needs.

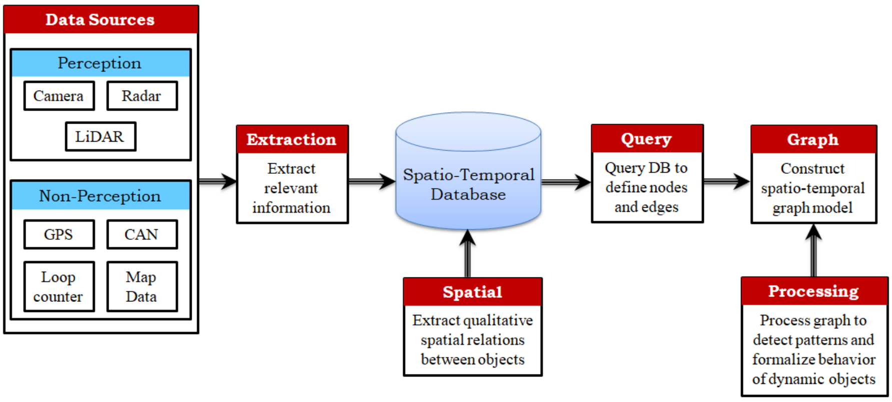


Figure 1.1: Global workflow for generating spatio-temporal graph to represent traffic

- **Extraction** module is responsible for extracting relevant information from raw data collected using different sensors. This module represents existing algorithms developed for processing images, videos, point clouds as well as non-perception vehicle and traffic data. These algorithms detect different object classes and pertinent data about each class. For the scope of this thesis, the choice of such off-the-shelf algorithms does not matter, however, they should be compared according to criteria like performance, type of data processed, type of information computed etc.

- Once the meaningful information has been computed by the previous module, it needs to be stored into a **Spatio-Temporal Database** while taking into account the time of acquisition (timestamp) of the data. Initially, a relational database can be considered to store this information with tables defined according to different classes of objects detected.

- To extract qualitative spatial relations between detected objects, **Spatial** module adds the required functionality to the database. An example of such module is PostGIS framework which represents objects according to different geometrical primitives and extracts spatial relations between them.

- **Query** module is responsible for querying the spatio-temporal database to define nodes and edges of the graph. Nodes are the objects which are detected by the **Extraction** module. These objects have some attributes according to the class to which they belong. These attributes have time-varying values and they are added as node labels for their corresponding nodes. Similarly, spatial relations are considered as edge labels having time-varying values.

- **Graph** module takes into account these nodes and edges and forms the graph. This module is responsible for adding temporal dimension within the graph to model both Topological and Attribute graph evolutions. The changes in graph structure and node and edge attributes are considered as *events* connecting two graph *states*.

- Once the Spatio-Temporal Graph is constructed, algorithms which exploit its structure and spatial relations between various entities can be applied, taken care of by **Processing** module, for various tasks. One such application of Traffic Pattern Detection is discussed in this thesis.

The mathematical models of traffic have been applied for detecting traffic patterns describing variations in traffic density, travel time, traffic congestion etc. Such patterns are referred to as *statistical patterns* since they are based on numerical description of traffic given using traffic data. Since traffic model proposed in this thesis formalizes the underlying structure of traffic, detecting patterns, referred to as *structural patterns*, taking into account various traffic constituents, provides a novel point of view to traffic pattern detection. Another major contribution of this thesis is the algorithm proposed to detect structural traffic patterns, represented using time-varying graphs, in the proposed spatio-temporal graph. This post-processing algorithm highlights one of the tasks carried out by **Processing** module in the workflow described above. The proposed algorithm considers the problem of pattern detection as *subgraph isomorphism* for time-varying graphs. This notion of detecting structural traffic patterns using the spatio-temporal graph demonstrates an important application of combining Traffic Modeling with the concepts developed in Spatio-Temporal Modeling against the backdrop of Graph Theory. Hence, as shown in Figure 1.2, this thesis aims to bring together the research from Traffic Modeling, Spatio-Temporal Modeling and Pattern Detection.

The manuscript is organised as follows:

- Chapter 2, entitled State of the Art, discusses existing research conducted for Traffic Modeling. Different point of views and scales of representation for traffic modeling are discussed. Then the chapter goes on to describe the existing work in Spatio-Temporal Modeling to bring forth the background of this field of research. Then the role that graphs have played in representing spatio-temporal phenomenon is discussed. The aim of this chapter is to motivate the notion of representing traffic using spatio-temporal graph.

- Chapter 3, entitled Spatio-Temporal Graph Model of Road Traffic, describes in detail the proposed model. First, the types of object classes, considered in our model as nodes, are presented. Then, various kinds of spatial relations considered between these

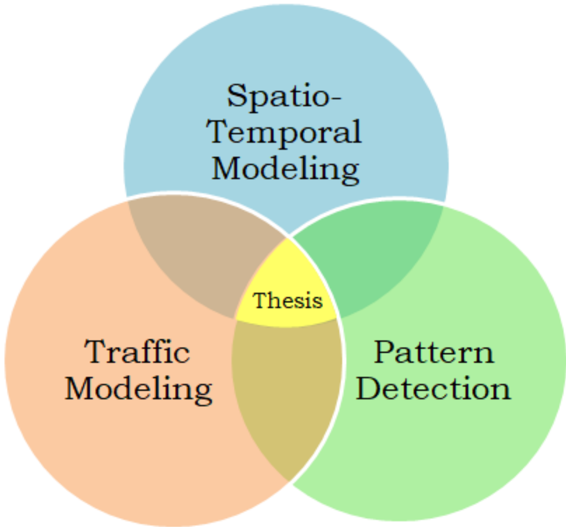


Figure 1.2: Research domains which this thesis brings together

objects are described. These relations are embedded into the edges of the graph. For different kinds of relations, the entities are modeled using different kinds of spatial primitives and the relations are defined from different frames of reference. After describing these concepts, the graph is formalized and different point of views of modeling traffic are discussed. Then, time is considered in the graph and its time varying characteristic is discussed. After presenting some related work on time-varying graphs, we go on to describe temporal concepts in our model. We discuss functions for node and edge presence and labeling which pave the way for two kinds of evolutions of graph - Topological (structure-based) and Attribute (node and edge attribute based).

- Chapter 4, entitled Structural Pattern Detection in Road Traffic, describes the application of the proposed model to detect structure-based traffic patterns. First, we discuss some related work for Traffic Pattern Detection and Graph-based Pattern Detection and describe the problem of subgraph isomorphism for static and time-varying graphs. Then some examples of kinds of structural patterns which could be detected in traffic if it is represented using a time-varying graph are described. These patterns represent simple traffic situations using time-varying graphs. Then the algorithm proposed to detect the said patterns in the spatio-temporal graph for traffic is presented. In the end, results for algorithm benchmarking, collected by applying the algorithm to randomly generated graphs using different set of parameters are discussed. These results show the effects of graph parameters on the time taken by the algorithm for detecting patterns.

- Chapter 5, entitled Conclusion and Perspectives, concludes the thesis and puts forth some future research directions which could be taken on the basis of the described work.

Note: To help with finding references included in the thesis, we have added the first author's last name, year of publication and the title of the reference as a side note when the reference is cited for the first time in each chapter. In numerical version of the manuscript, by clicking on the cited reference, the reader can find it listed in the Bibliography and by clicking on the title of the reference in the side note, it can be located on the web.

2

State of the Art

Since in this thesis we will propose a traffic model, let us start our discussion by describing some existing research from the field of Traffic Modeling. In this chapter, we will present different ways to model traffic and discuss different representation scales at which traffic can be described. Then we will move on to present some existing research for the field of Spatio-Temporal Modeling as our ultimate aim is to show that since road traffic is spatio-temporal, the models developed to represent spatio-temporal phenomenon can be applied to traffic. Out of various modeling techniques which have been proposed for spatio-temporal models, modeling traffic using spatio-temporal graphs is the most suitable for our case. This intuition is realized towards the end of this chapter.

2.1 MODELING ROAD TRAFFIC

Vehicular traffic modeling has been an attractive domain for various research communities for the past several decades. One of the reasons for such a keen interest is the effect road traffic has on the society and daily lives of people around the world. After the end of the second world war, a need for setting up a public research authority was felt whose objective would be to conduct research on every aspect of traffic flow. An excerpt taken from Aldington, Beaumont, and Nicholl, 1945[1] (page 365):

> Mr. W. P. Robinson: ... Before the war we tried to find out the volume of traffic by a triennial census, and we purposely avoided taking that census at peak times of traffic. I suggest that that was quite wrong, and that what we require now is a proper Government traffic research authority which will explore every detail of traffic flow, speed, and volume, and, in addition, investigate the way it does, or does not, observe the laws of the land. ...

This lead to an increased interest towards formalizing theoretical and experimental aspects of traffic flow. Wardrop, 1952[2], in his paper, pushed for developing theoretical methods which, he argued, are necessary for the advancement of otherwise empirical research domain of traffic engineering. Despite his best efforts of introducing theoretical methodologies, he ended up introducing statistical quantities such as speed distribution in space-time and the rate of overtaking in a traffic

[1] *Aldington et al. (1945). The Post-War Development of Road Motor Transport*

[2] *Wardrop (1952). Road Paper. Some Theoretical Aspects Of Road Traffic Research.*

stream. A different theoretical approach for understanding vehicular traffic as a whole was proposed by Lighthill and Whitham, 1955[3], who employed the hypothesis that vehicles in traffic are similar to particles in fluid, and proved the existence of kinematic waves in traffic moving on a long road in one direction. However, this approach was limited to cases when traffic is considered as a continuous single entity [4].

[3] *Lighthill et al. (1955). On kinematic waves II. A theory of traffic flow on long crowded roads*

[4] *See Section 2.1.3.2*

2.1.1 *Vehicular traffic as thermodynamic system*

Kinetic theory for vehicular traffic

Over the years, an enormous amount of research was published which applied the analogy between thermodynamic systems and vehicular traffic. One of the noteworthy ideas, which got a lot of attention and was subject to some controversies in the research community, was that of applying Boltzmann-like equation to explain spatio-temporal evolution of vehicular velocity distributions and was proposed in Prigogine and Herman, 1971[5]. It was argued that Prigogine-Boltzmann equation did not reflect the real-world traffic flow since correlation between nearby drivers was neglected. To remedy this, an improved Boltzmann-like model for traffic flow was proposed in Paveri-Fontana, 1975[6]. Nevertheless, the kinetic theory for vehicular traffic was extended from highway traffic to intra-city traffic.

[5] *Prigogine et al. (1971). Kinetic Theory of Vehicular Traffic*

[6] *Paveri-Fontana (1975). On Boltzmann-like treatments for traffic flow*

Two-fluid model

Prigogine, Herman and their coworkers performed various experiments in different cities in U.S. and found that trip (or travel) time per unit distance was linearly related to stop time per unit distance. This lead them to propose a theoretical model in Herman and Prigogine, 1979[7], called Two-fluid model, which considered road traffic to consist of two "traffic fluids" - one consisting of moving cars and the other of stopped cars (excluding parked cars).

[7] *Herman et al. (1979). A Two-Fluid Approach to Town Traffic*

Multi-lane traffic model

In addition to implicitly modeling different lanes on a highway, as is done in the models mentioned above, there has also been an interest to consider multiple lanes explicitly. A Boltzmann-like multi-lane traffic model based on the improved kinetic theory of vehicular traffic from Paveri-Fontana, 1975 was proposed in Helbing and Greiner, 1997[8]. The authors take into account the interactions between vehicles moving in neighbouring lanes in terms of change in velocities of faster vehicles due to the presence of slower ones in their vicinity.

[8] *Helbing et al. (1997). Modeling and simulation of multilane traffic flow*

Multi-class traffic model

Traffic models mentioned above considered vehicular traffic to be homogeneous, in the sense that different vehicle classes like cars, trucks etc. were not distinguished. Hoogendoorn and Bovy, 2000[9] argue that distinguishing vehicle classes and incorporating class-specific behaviour would lead to more accurate traffic models. They extend single-class *gas-kinetic* equations proposed in Paveri-Fontana, 1975 to multi-class case and take into account the effects of inter-class as well as intra-class vehicle interactions. This approach is also investigated in Wong and Wong, 2002[10] and the merits of considering road traffic consisting of heterogeneous users are highlighted by showing that some traffic flow phenomenon are better explained if different vehicle classes are explicitly considered.

[9] *Hoogendoorn et al. (2000). Continuum modeling of multi-class traffic flow*

[10] *Wong et al. (2002). A multi-class traffic flow model - an extension of LWR model with heterogeneous drivers*

Multi-lane multi-class traffic model

Hoogendoorn and Bovy, 2001b[11] combines the notions of considering traffic to be multi-lane and heterogeneous (multi-class) and proposes a model reflecting the behaviour of real-world road traffic. It extends previous models to take into account the user-specific and class-specific interactions between vehicles.

[11] *Hoogendoorn et al. (2001). Platoon-Based Multiclass Modeling of Multilane Traffic Flow*

2.1.2 *Vehicular traffic as (vehicle)ular traffic*

When modeling vehicular traffic, one should not forget about vehicles themselves. Following this, a novel point of view for representing vehicular traffic received a lot of research interest in last decades. In this case, behaviour of individual vehicles and drivers was the main focus using which the aggregated characteristics of traffic were computed.

Car-following models

A pioneer work which lead the research in this direction was that of Pipes, 1953[12]. It considers two cases - when a line of vehicles starts moving from standstill and when the line of vehicles already in motion comes to a halt - and relates the velocities of the trailing vehicles with the velocity of the leading one in both. The author assumes that all vehicles keep a safe distance between themselves which increases linearly with velocity. However, the model doesn't include the driver's response time to changing traffic conditions. An improvement to this model is proposed in Chandler, Herman, and Montroll, 1958[13] where, driver's reaction time, after perceiving a change in traffic condition in front, is also taken into account. The model considers the velocity difference between leading and following vehicle as the stimulus leading to driver's response, occurring after some finite delay. This model is further extended in Gazis, Herman, and Potts, 1959[14], which

[12] *Pipes (1953). An Operational Analysis of Traffic Dynamics*

[13] *Chandler et al. (1958). Traffic Dynamics: Studies in Car Following*

[14] *Gazis et al. (1959). Car-Following Theory of Steady-State Traffic Flow*

adds the effect of relative distance between leading and following vehicle on driver's sensitivity. A general equation of driver's sensitivity and its application in computing aggregated traffic quantities like flow and concentration were discussed in Gazis, Herman, and Rothery, 1961[15], and it was shown that sensitivity differs depending on traffic density. Various models which employed and extended the notions put forth by Gazis, Herman, and Rothery, 1961 were developed in later years, a comprehensive survey on which was conducted by Brackstone and McDonald, 1999[16].

[15] *Gazis et al. (1961). Nonlinear Follow-the-Leader Models of Traffic Flow*

[16] *Brackstone et al. (1999). Car-following: a historical review*

Cellular automata models

Towards the end of the last century, models describing vehicle movements in discrete space and time were proposed. They considered space to be divided into a grid of cells in which vehicles hopped from cell-to-cell in one time step according to their velocity. This idea was used to model road traffic in Nagel and Schreckenberg, 1992[17] (referred to as Nagel-Schreckenberg (NaSch) model). The authors defined update rules for individual vehicles to model their behaviour. A randomisation term was defined to take into account random change in vehicle speed (to model different driving behaviours). However, this randomisation term lead to the possibility of traffic jams even at lower traffic density. In addition, no theoretical basis for the randomisation rule was given.

[17] *Nagel et al. (1992). A cellular automaton model for freeway traffic*

Some cellular automata models considering multi-lane roads have also been proposed in the literature. Rickert et al., 1996[18] extend the stochastic single-lane NaSch model to the case of multi-lane roads and adds an extra update step for performing lane-change maneuver. The lane-change is performed if there are sufficient empty cells in the target lane. Nagel et al., 1998[19] generalize various multi-lane models proposed and compare the lane-change rules on German and American highways.

[18] *Rickert et al. (1996). Two lane traffic simulations using cellular automata*

[19] *Nagel et al. (1998). Two-lane traffic rules for cellular automata*

In addition to modeling highway traffic, cellular automata models have also been employed to model microscopic behaviour of urban traffic, in which case, two kinds of approaches could be applied. The first, called grid approach, considers a 2D cellular grid to represent urban road network, and was proposed in Biham, Middleton, and Levine, 1992[20]. Even though this model didn't represent real-world traffic, it was able to capture the occurrence of traffic jams at higher density. The second approach considers more realistic road network with separate single-lane or multi-lane roads connected using intersections. An example of such an approach is considered in Esser and Schreckenberg, 1997[21]. The proposed model is based on NaSch model and is able to simulate fairly complex structure of urban road network, including vehicle priority at road intersections.

[20] *Biham et al. (1992). Self-organization and a dynamical transition in traffic flow models*

[21] *Esser et al. (1997). Microscopic Simulation of Urban Traffic Based on Cellular Automata*

The idea of using cellular automata models for modeling vehicular traffic has been receiving a lot of interest due to their ease of implementation, even though, such models don't always provide very accurate picture of macroscopic traffic quantities. A survey on this modeling technique for road traffic modeling is provided in Maerivoet and Moor, 2005[22].

Sections 2.1.1 and 2.1.2 describe various traffic models from different point of views. Thermodynamics-based models consider traffic from an abstract view, with its aggregate variables being the focus of interest. Considering traffic in terms of individual vehicles requires more detailed view of traffic. In the literature, vehicular traffic is modeled at different representational scales, which we will discuss in the following Section.

[22] Maerivoet et al. (2005). *Cellular automata models of road traffic*

2.1.3 *Scale of representation in traffic modeling*

Vehicular traffic is a complex phenomenon, where complexity arises from the interactions among its constituents as well as due to the effect of infrastructural elements on their motion. Helbing, 2001[23] considers vehicular traffic to be a self-driven many-particle system, where the particles affect the behaviour of other particles in their vicinity. Modeling such systems by considering individual particles provides a detailed view of the dynamics of the whole system. However, such models have proven to be computationally expensive due the number of variables required to model the entire system, as noted in Hoogendoorn and Bovy, 2001a (Section 7)[24]. On the other hand, aggregated modeling of such systems could also describe their dynamics. However, in this case, the information about characteristics of individual particles is lost. This trade-off has lead to the development of plethora of models representing traffic at different scales. In this subsection, we will discuss three major categories of traffic models - microscopic, macroscopic, and hybrid (also called mesoscopic).

[23] Helbing (2001). *Traffic and related self-driven many-particle systems*

[24] Hoogendoorn et al. (2001). *State-of-the-art of vehicular traffic flow modelling*

2.1.3.1 *Microscopic Traffic Models*

Considering traffic as a collection of individual vehicles and modeling variation in quantities like vehicle position and velocity with time, falls under the category of microscopic modeling. The models belonging to this category trace vehicle characteristics over time which are used to compute aggregated traffic parameters like flow and density.

Main categories of models which represent traffic microscopically are Car-Following Models (like the one from Pipes, 1953, discussed in Section 2.1.2), Optimal Velocity Models (from Newell, 1961[25]), Cellular Automata models (like NaSch from Nagel and Schreckenberg, 1992, discussed in Section 2.1.2), and recently developed Intelligent Driver Models (proposed in Treiber, Hennecke, and Helbing, 2000[26]).

Some works which use microscopic models to compute aggregated equations relating traffic variables could be found in the literature. For example, the macroscopic model proposed in Payne, 1971[27], describing

[25] Newell (1961). *Nonlinear Effects in the Dynamics of Car Following*

[26] Treiber et al. (2000). *Congested traffic states in empirical observations and microscopic simulations*

[27] Payne (1971). *Model of freeway traffic and control*

the relation between vehicle velocity and traffic density, is based on the car-following model from Newell, 1961. Particle hopping models, like Cellular Automata NaSch model, have also been proven capable, despite their simplistic implementation, to describe fluid-dynamic behaviour observed in macroscopic models in Nagel, 1998[28]. In addition, mesoscopic models could also be derived from microscopic view of traffic, as described in Klar and Wegener, 1998.

28 Nagel (1998). *From Particle Hopping Models to Traffic Flow Theory*

2.1.3.2 *Macroscopic Traffic Models*

Macroscopic models consider traffic as a continuous aggregated entity and describe the spatio-temporal variation of average quantities like traffic density, traffic flow and average speed. Since traffic is considered from a coarser point of view, the interactions between individual vehicles for describing traffic situations are ignored.

The pioneer work towards the development of macroscopic models, which considers first-order fluid-dynamic approximation of road traffic, is from Lighthill and Whitham, 1955. It models variation of flow, density and speed under steady-state traffic conditions, and make a simplified assumption that average speed at a location and at a given time instant varies instantaneously with traffic density at that location. This assumption leads to some unrealistic behaviour in vehicle acceleration, as noted in Papageorgiou, 1998[29]. These models, called Lighthill-Whitham-Richards (LWR) models in the literature, were further improved by a second order model from Payne, 1971, which takes into account the driver's reaction time while modeling average speed.

29 Papageorgiou (1998). *Some remarks on macroscopic traffic flow modelling*

Combining macroscopic and microscopic modeling methodologies emphasizes on demonstrating the applicability of macroscopic modeling theory in deriving microscopic vehicle characteristics. For example, Castillo, 1996[30] derives a linear car-following model proposed by Chandler, Herman, and Montroll, 1958 and its non-linear version, using LWR macroscopic model. Recently, to model lane-change (a microscopic behaviour), Laval and Leclercq, 2008[31] proposed to apply discrete-time version of macroscopic lane-changing model.

30 Del Castillo (1996). *A car following model-based on the Lighthill-Whitham theory*

31 Laval et al. (2008). *Microscopic modeling of the relaxation phenomenon using a macroscopic lane-changing model*

2.1.3.3 *Mesoscopic Traffic Models*

Due to the trade-off between high computation cost of microscopic models and a coarser aggregated description of traffic by macroscopic models, a middle ground is reached by combining the above mentioned modeling approaches, giving rise to the so-called mesoscopic (or hybrid) models. Even though we are using terms "mesoscopic" and "hybrid" interchangeably, there is a subtle difference between them. Although both types of models have as objective the combination of macroscopic and microscopic models to overcome their collective shortcomings, the way in which these approaches are combined differs.

Mesoscopic models consider both macroscopic and microscopic characteristics of traffic to co-exist. For example, the gas-kinetic models, like the one from Paveri-Fontana, 1975, discussed in Section 2.1.1, consider velocity distribution of individual vehicles along with interactions between them, without tracing the velocity variation, over time, for each vehicle. Similarly, Hoogendoorn and Bovy, 1998[32] proposed a model describing the distribution of time-headway (time elapsed between passing, of two successive vehicles, a point on the road) for each vehicle, without considering individual vehicle dynamics.

[32] *Hoogendoorn et al. (1998). New Estimation Technique for Vehicle-Type-Specific Headway Distributions*

On the other hand, hybrid models consider microscopic and macroscopic characteristics to co-exist but at different locations, with specific attention given at boundaries or interfaces of two representations. The objective here is to benefit from both modeling approaches for explaining different kinds of traffic phenomenon in a single model. This notion was first introduced in Bourrel and Henn, 2002[33], where a hybrid model based on LWR approach was proposed. At the interface of two representations, it was assumed, that both approaches co-exist, similar to mesoscopic models.

[33] *Bourrel et al. (2002). Mixing micro and macro representations of traffic flow*

The field of Traffic Modeling has been the center of research for past several decades. Due to the consideration that traffic is analogous to fluids, physicists have developed various traffic models by applying the concepts of fluid dynamics. The models which consider individual vehicles mainly apply the laws of motion to describe their behaviour.
The main interest for Traffic Modeling has been to relate traffic variables like flow, density and speed, whether traffic is considered as aggregated entity or in terms of individual constituents. Moreover, the interactions between vehicles are modeled in terms of change in speed or probability of lane change due to the presence of slow moving vehicles.
Towards this end, proposed models take a quantitative approach since they only rely on data collected at a given point in space-time. Due to the amount of data required to model traffic, either the representation scale of traffic is varied or discrete models like cellular automata are proposed.
In this thesis, we will see if we can model traffic in qualitative terms, i. e. if using quantitative data, some qualitative knowledge can be derived which could help to comprehend the evolution of traffic from a coarser point of view.
The field of Spatio-Temporal Modeling, discussed in the next section, deals with modeling spatio-temporal phenomenon from both quantitative and qualitative standpoints. Let us see if some concepts from this field can be applied in case of road traffic using which a spatio-temporal model of traffic can be developed.

2.2 SPATIO-TEMPORAL MODELING

Phenomena vs noumena

Philosopher Immanuel Kant, in his book titled *Critique of Pure Reason*, first published in 1781, says that knowledge can be acquired through experiencing and observing the things around us. However, we are limited by our senses while doing so, hence, we only perceive the appearances of those things and not the things-in-themselves (independent of our perception). He refers to the things that we observe (or are capable of observing due to our limitations) as *phenomena* and things-in-themselves as *noumena*. The notion of phenomenon can be extended to define spatio-temporal phenomenon, as something which is observed or can be observed over space and time.

The interest to comprehend and exploit the evolution of a phenomenon, with space and time, has lead the researchers to propose models which represent, and store data about, such phenomenon. This resulted in the birth of a new research domain, a sub-domain of Artificial Intelligence, called Spatio-Temporal Data Modeling. As noted in Tsichritzis and Lochovsky, 1977[34], the objective behind defining a data model is to establish some guidelines for the representation of entities which take part in the phenomenon to be modeled as well as the relationships between those entities. An entity is defined as the "thing" of interest about which data is or will be stored in the database, as noted by Chen, 1976[35]. It may be a person, an event, an organisation etc.

[34] *Tsichritzis et al. (1977). Data base management systems*

[35] *Chen (1976). The Entity-relationship Model*

Identity of entities

Considering distinct entities to be present in the world, the problem of identifying them and their associated attributes in the database becomes pertinent. The task of identifying an entity is performed by defining a unique *identity* for that entity. The importance of defining the identity of an entity in a database is described in Khoshafian and Copeland, 1986[36]. It is different from all its other attributes since it represents the uniqueness of that entity. For example, if the database considers cars as entities, their attributes like manufacturer, mileage, horsepower, seating capacity etc. could have same values for different entities and hence can't be used to uniquely identify them. However, chassis number or vehicle identification number, which is unique for every car, could act as the identity for each car (entity) in the database.

[36] *Khoshafian et al. (1986). Object Identity*

Spatial data models

Spatial data models could be differentiated from non-spatial data models in terms of the type of entities they consider and relationships between them. Such models are used to represent spatial phenomenon which involve describing spatial entities with spatial relations between them, as noted by Peuquet, 1984[37]. Detailed description about spatial entities and relations is given in Section 2.2.1.1. Spatial models describe continuous real world which can be abstracted according to the needs of the application. Goodchild, Haining, and Wise, 1992[38] point out that abstraction required to conceive a digital model from reality plays an important role in determining the complexity of the model and the amount of information that the model represents.

37 Peuquet (1984). *A Conceptual Framework and Comparison of Spatial Data Models*

38 Goodchild et al. (1992). *Integrating GIS and spatial data analysis: problems and possibilities*

Temporal data models

Modeling continuously varying real-world phenomenon requires the incorporation of temporal knowledge into the model. The importance of time in data modeling can be understood from the fact that Clifford and Warren, 1983[39] pushed for the development of models having temporal dimension as an inherent part of the model itself. Doing this, it was argued, would lead to a consistent modeling of variations in the phenomenon being modeled as past data would not be "forgotten". Over the years, various temporal data models were proposed. A noteworthy work in this direction was from Snodgrass and Ahn, 1986[40] where three different kinds of time - valid, transaction and user-defined - were considered. Valid time refers to the time at which the information modeled in the database was or will be true or valid. Transaction time, on the other hand, refers to the time at which the data about the real world was added in the database. User-defined time is used to model additional temporal information not modeled with valid or transaction time. Models including both, valid and transaction time, are referred to as bi-temporal models, as described in Jensen, Soo, and Snodgrass, 1994[41]. For a detailed description about time, see Section 2.2.1.2.

39 Clifford et al. (1983). *Formal Semantics for Time in Databases*

40 Snodgrass et al. (1986). *Temporal Databases*

41 Jensen et al. (1994). *Unifying temporal data models via a conceptual model*

Spatio-temporal data models

Combining approaches of spatial and temporal data modeling leads to the development of spatio-temporal models. Sinton, 1978[42] points out that if we want to extract useful information from an observation recorded in the database, three components - theme, location and time - about that observation should be taken into account. "Theme" describes the phenomenon or object being observed *(what)*. "Location" describes where the observation was recorded *(where)*. "Time" represents when the phenomenon or object was observed *(when)*.

Although time was included in data models before, considering time in spatial data models gained momentum in late 1980s. An introductory work in this direction was from Langran and Chrisman, 1988[43], where, as in Snodgrass and Ahn, 1986, the importance of valid

42 Sinton (1978). *The inherent structure of information as a constraint to analysis*

43 Langran et al. (1988). *A Framework For Temporal Geographic Information*

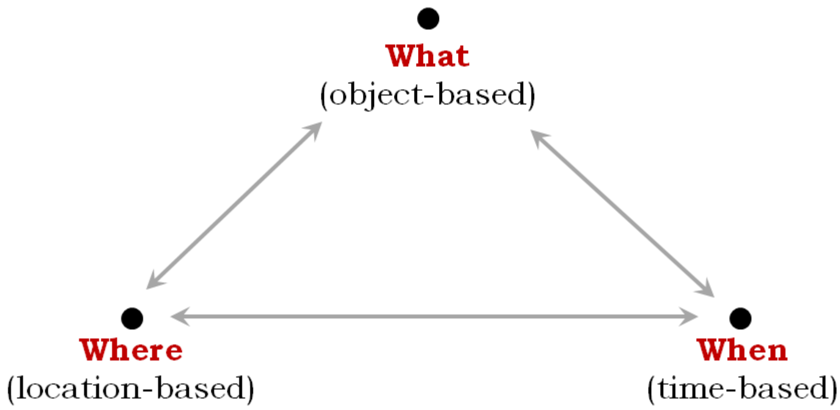


Figure 2.1: Triad framework as described in Peuquet, 1994

and transaction time was emphasised. In addition, authors argue that spatial and temporal dimensions could be considered similar to each other and, hence, methods developed for spatial modeling could be extended easily to include time.

However, spatial and temporal dimensions cannot always be considered similar. Peuquet, 1994[44], while looking at space-time from philosophical and scientific point of views, mentions that there are underlying differences between space and time and spatio-temporal models should appreciate these differences and consider an integrated approach while describing a real-world phenomenon. The author proposed an integrated spatio-temporal framework, which incorporates three types of representations - object-based *(what)*, location-based *(where)* and time-based *(when)* - similar to Sinton, 1978, which is referred to as Triad framework (Figure 2.1).

[44] Peuquet (1994). *It's About Time: A Conceptual Framework for the Representation of Temporal Dynamics in Geographic Information Systems*

With a brief introduction to the domain of Spatio-Temporal Modeling, let us now delve into the underlying concepts of such models - space and time - and see different point of views of including them.

2.2.1 *Concepts of space and time*

2.2.1.1 *Space*

Absolute vs. relative space

Historically, space has been considered from two point of views - absolute and relative. Absolute point of view, employed by Newton, considered space as an immutable void, containing distinct objects, movement or variation of which, has no effect on space itself. It considers space separately from objects it contains. Relative view, however, advocated by Leibniz, does not consider space as being separate but as an integral characteristic of those objects. With this point of view, relations between objects are necessary to define the notion of space. Even though these point of views have been debatable and contradictory, as described in Chrisman, 1977[45], Peuquet, 1994 argues that, in fact, these two considerations are complimentary.

[45] Chrisman (1977). *Concepts of space as a guide to cartographic data structures*

Basic units of space

Following these two point of views, Chrisman, 1977 considers two basic units of space - places and objects. In former case, space consists of collection of discrete places, forming a grid-like structure. Since, in this case, space is considered separate from objects, it is possible to have a grid structure which doesn't contain any objects. Such representation of space is referred to as *raster* representation, where each "place" has some attributes associated to it.

Considering objects as distinct units of space results in a point of view where space is characterized by relations between these objects. To define object structure, geometrical primitives like points, lines and polygons are used. Such representation is called *vector* representation of space, and, in this case, the objects are associated with one or more attribute(s), with their location being one of them.

These two point of views of space have led Peuquet, 1988[46] to propose a dual framework with the aim of integrating their merits. Location-based representation in Peuquet's dual framework corresponds to "places" being the basic units of space in Chrisman's model, and object-based representation in dual framework corresponds to, well, objects being the basic units of space.

[46] Peuquet (1988). *Representations of Geographic Space: Toward a Conceptual Synthesis*

Spatial entities

Depending on two point of views described above, Peuquet, 1988 argues that spatial entities could either be objects (object-based representation) or locations (location-based representation). Objects such as a road segment, a land parcel, a vehicle etc. could be considered as spatial entities, to which properties or attributes can be associated. In this representation, relations between these objects need to be defined explicitly. On the other hand, when locations are considered as entities, attributes are associated to each location and implicit relations between two locations are considered.

Representation of spatial entities

Having defined the notion of spatial entity, the question that comes to mind is "how to represent spatial entities in the spatial data model?". The answer to this question lies in the spatial representation chosen. For vector representation, where individuality of objects is given importance, distinct objects are represented using geometrical primitives. As Couclelis, 1992[47] points out, this representation is more suited for modeling artificial or man-made objects like roads, buildings, and administrative land boundaries. In addition, relations between simple geometrical primitives could be considered to exist which form

[47] Couclelis (1992). *People manipulate objects (but cultivate fields)*

complex geometries (like relations between points forming a line and related lines forming a polygon).

In contrast, raster representation of space is preferred to model the natural world. It has been widely used to model spatially continuous variables like temperature, atmospheric pressure etc. using the notion of *field*. Galton, 2001[48] defines field as a function which maps distinct locations with values of attributes under consideration at that location. Hence, the spatial extent of the field is given as set of locations it overlaps. In theory, field varies continuously over space. However, to model it in digital information systems, it has to be discretized in terms of space it occupies. Different methods of discretizing space lead to different field models, as noted in Goodchild, 1993[49].

[48] Galton (2001). *A Formal Theory of Objects and Fields*

[49] Goodchild (1993). *The state of GIS for Environmental Problem-Solving*

The debate regarding raster vs. vector representations and objects vs. fields modeling primitives has been addressed by various authors over last decades. However, since field-based approach is out of scope of the present thesis, this topic is not discussed further, while concepts related to object-based modeling are given importance since they are utilized in the model proposed in this thesis.

Object-based modeling

In case of object-based modeling, spatial relations between object geometries are required to understand the concept of space. Such relations have been defined in the literature and surveyed in Chen et al., 2013[50]. In this thesis we focus on qualitative binary relations, i. e. relations defined between geometries of two objects.

[50] Chen et al. (2013). *A survey of qualitative spatial representations*

Spatial relations: topological

Perhaps the most common category of spatial relations studied in the literature is *topological* relations. These relations are qualitative in nature and remain invariant under topological changes, like translation, rotation and scaling, in object geometry. Considering 2D space containing objects represented using regions (polygons), Randell, Cui, and Cohn, 1992[51] developed a logic capable of handling different degrees of connection between two such regions. A general connectivity relation, $C(x, y)$, meaning 'x is connected to y', for two 2D regions x and y, is used to define 8 different topological relations: disconnected (DC), externally connected (EC), partial overlap (PO), tangential proper part (TPP) with its inverse (TPPI), equal (EQ) and non-tangential proper part (NTPP) and its inverse (NTPPI), as shown in Figure 2.2. This formalism is referred to as RCC8.

[51] Randell et al. (1992). *A Spatial Logic Based on Regions and Connection*

Another noteworthy formalism for defining topological relations is proposed in Egenhofer and Herring, 1991[52]. The representation, called 9-Intersection Matrix (9-IM), defines relations between any combination of points, lines and polygons by considering their interior, exterior and boundary separately. For two spatial primitives x and

[52] Egenhofer et al. (1991). *Categorizing binary topological relations between regions, lines, and points in geographic databases*

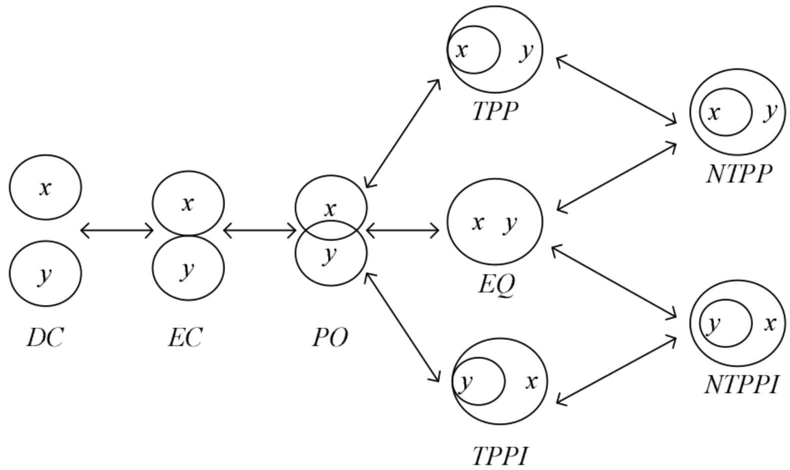


Figure 2.2: Topological relations defined using RCC8. Under continuous movement or deformation of regions, relation next to a given relation in the direction of the arrows is considered. Figure is taken from Chen et al., 2013.

y, nine fundamental intersections between their interiors (x° and y°), exteriors (x^- and y^-) and boundaries (∂x and ∂y) are defined using a 3x3 intersection matrix. For example, relation DC defined using RCC8 between two regions could be given as:

$$\begin{array}{cc} \begin{matrix} y^\circ & \partial y & y^- \end{matrix} & \\ \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix} & \begin{matrix} x^\circ \\ \partial x \\ x^- \end{matrix} \end{array}$$

with 1's representing the parts which have topological relations and 0's representing un-related parts of both x and y.

Spatial relations: orientation

A different category of spatial relations is *orientation* relations which describe where an object is placed relative to another object. These relations have three components: target object, reference object and frame of reference. In 2D space, with frame of reference being at the same position as reference object, Hernández, 1993[53] defines 8 orientation relations: front, back, left, right, left-back, left-front, right-back and right-front, by distinguishing the space around an object into 8 parts. However, changing the orientation of the reference frame itself, leads to change in relative orientation between target and reference objects. This point becomes clear from the formalism proposed in Freksa, 1992b[54]. In general, orientation relations are defined between points as geometrical primitives and this notion is followed in this thesis.

[53] *Hernández (1993). Maintaining qualitative spatial knowledge*

[54] *Freksa (1992). Using orientation information for qualitative spatial reasoning*

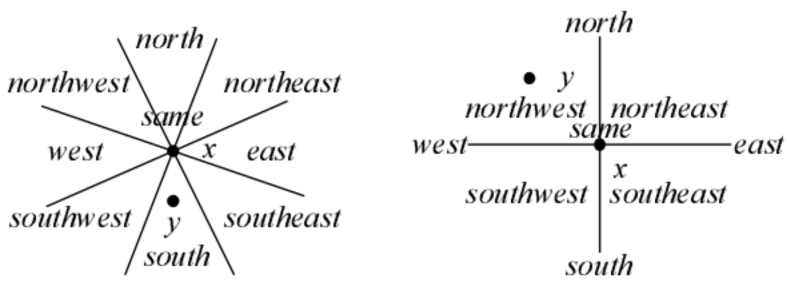


Figure 2.3: Cone-shaped and projection-based methods to describe cardinal directions. Figure is taken from Chen et al., 2013.

Spatial relations: direction

A related category of spatial relations is *direction* relations. While orientation relations make sense in small-scale space (space within perception from a single vantage point), direction relations like north, south-east etc. are preferred in large-scale space (beyond perception from single vantage point) to describe relative locations of spatial objects. Frank, 1996[55] describes two methods to formalize direction relations: cone-shaped and projection-based, shown in Figure 2.3, and points out that projection-based system is more accurate than cone-shaped. Furthermore, in case of direction relations, the frame of reference remains invariant and is external to both target and reference objects. The importance of frame of reference and its effect on the derivation of orientation and direction relations are discussed in Clementini, 2013[56].

[55] *Frank (1996). Qualitative spatial reasoning: cardinal directions as an example*

[56] *Clementini (2013). Directional relations and frames of reference*

Spatial relations: distance

In addition to direction and orientation relations between objects, *distance* between them is required to accurately describe their relative positions. Even though, distance provides metric information which is quantitative, qualitative distance information is more intuitive from cognitive point of view. Clementini, Felice, and Hernández, 1997[57] describes distance in qualitative terms like very close, close, very far etc. and relates them to quantitative distance measures.

[57] *Clementini et al. (1997). Qualitative representation of positional information*

Spatial relations: linear order

Considering distinct spatial entities existing in 1D space, *linear order* relations like before and after, can be defined between them. In this case, frame of reference is external to all entities. Here, 1D space is thought of as a projection of 2D space on a linear dimension. Linear order relations have been described in Guesgen, 1989[58].

[58] *Guesgen (1989). Spatial reasoning based on Allen's temporal logic*

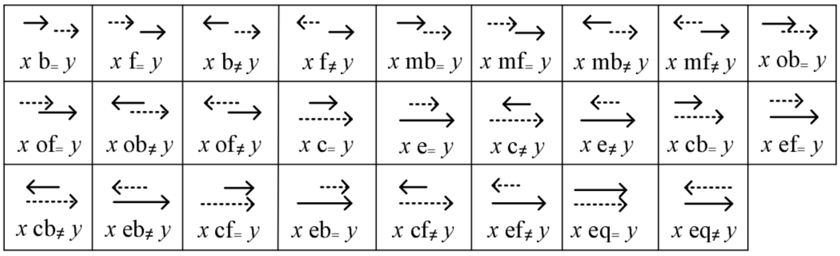


Figure 2.4: Relations defined using Directed Interval Algebra. The symbols b, f, m, o, c, e, eq stand for behind, in-front-of, meets, overlaps, contained-in, extends and equals respectively. Figure is taken from Chen et al., 2013.

Spatial relations: relative trajectory

In addition to spatial relations between static objects, some spatial relations between moving objects could be found in the literature. Renz, 2001[59] developed an algebra, using 1D spatial intervals, to describe relations between moving objects. The algebra, called Directed Interval Algebra (DIA), attaches a direction vector to spatial intervals and proposes 26 relations between such intervals, shown in Figure 2.4. Intervals in same direction are represented using ($=$) and opposite direction intervals using ($\neq$). This algebra is suitable to define *relative trajectories* of moving objects.

[59] *Renz (2001). A Spatial Odyssey of the Interval Algebra: 1. Directed Intervals*

Spatial relations: relative speed

Another approach, called Qualitative Trajectory Calculus (QTC), proposed by Weghe, Cohn, and Maeyer, 2004[60], compares trajectories of moving points instantaneously, in terms of direction of movement and speed. At a given time instant, direction of movement of a point towards another point is represented using ($-$), direction of movement away from a point is represented using ($+$), and no movement is represented using (0). The *relative speed* is represented using the same symbols with ($-$) meaning slower, ($+$) meaning faster and (0) meaning same speed.

[60] *van de Weghe et al. (2004). A Qualitative Representation of Trajectory Pairs*

2.2.1.2 *Time*

Change

We live in a ever-changing and evolving world, and to understand this evolution we need to incorporate temporal knowledge into our formalisms. But before talking about time itself, let us look at what constitutes "change" and different ways it has been considered in the past.

Russell, 1903[61] defines change as follows:

> ...Change is the difference, in respect of truth or falsehood, between a proposition concerning an entity and a time T

[61] *Russell (1903). Principles of Mathematics*

> and a proposition concerning the same entity and another time T', provided that the two propositions differ only by the fact that T occurs in the one where T' occurs in the other....

In other words, change can be quantified by the difference in the value of a proposition at times T and T', and if the difference equals zero, it means that no change occurred. He goes on to say that:

> ...Change is due, ultimately, to the fact that many terms have relations to some parts of time which they do not have to others...

In a way, notion of change is necessary to even think about time, as noted by Shoham, 1985[62].

Change can be considered discrete or continuous. It is said to be discrete when, following the occurrence of an event or performance of an action, the state s of the world transforms into a new state s'', with no knowledge about, or of the existence of, intermediate state s', lying between s and s''. This idea of discrete change is modeled within Situation Calculus, proposed by McCarthy and Hayes, 1969[63], where current situation of the world changes (discretely) after an action or set of actions is performed. From a different perspective, occurrence of a event e can lead to change in the value of some property of an entity, which had a certain value in the time period before e occurred. Defining the world in terms of events was described in Event Calculus, proposed in Kowalski and Sergot, 1986[64]. However, even in this case, change in the value of the property was considered in discrete sense.

Change is said to continuous if answers to questions like "how the action was performed?" or "what happened during that event?" are known. That means, the duration of occurrence of action or event needs to be taken into account. To formalize a general theory of change, Shoham, 1985 notes that the developed theory be able to handle continuous changes since most processes in real-world are continuous. The idea to model such elongated processes to explain gradual changes leading to a new state of the world has been described in Hendrix, 1973[65]. The author also gives a hint that depending on the considered level of detail, change can be either discrete or continuous.

General concept of change can be modified depending on the application and research domain. For example, in case of qualitative temporal reasoning, Freksa, 1992a[66] describes gradual changes occurring in temporal relations between intervals. In the domain of information systems, change can be defined when an existing entity ceases to exist or a new entity is created having identity of currently non-existent entity. To describe change in entity and its identity in an information system, a change descriptive language is developed in Hornsby and Egenhofer, 1997[67].

[62] Shoham (1985). *Ten requirements for a theory of change*

[63] McCarthy et al. (1969). *Some Philosophical Problems from the Standpoint of Artificial Intelligence*

[64] Kowalski et al. (1986). *A logic-based calculus of events*

[65] Hendrix (1973). *Modeling simultaneous actions and continuous processes*

[66] Freksa (1992). *Temporal reasoning based on semi-intervals*

[67] Hornsby et al. (1997). *Qualitative representation of change*

Absolute vs. relative time

Just like space, time has also been considered from two point of views - absolute and relative, which are compared by Lin, 1991[68]. Absolute theory of time, similar to absolute theory of space, considers time to be a well-defined quantity with no relation to anything external. In this case, time is considered linear and unbounded, and is represented as collection of events occurring at different time instants, having no effect on time itself. On the other hand, relational view of time postulates that the existence of time cannot be explained without considering time as a characteristic of events. This point of view is used to describe relations between events which occur during finite intervals. Events can be considered analogous to objects in relational view of space. The question whether time consists of instants (absolute theory) or intervals (relative theory) has been discussed from philosophical point of view by Hamblin, 1972[69].

[68] *Lin (1991). Two theories of time*

[69] *Hamblin (1972). Instants and Intervals*

For most people, philosophers or not, absolute view of time, with time being linear and unbounded, and time instants being its primitives, provides a more natural way of picturing time. The representation provided by Situation Calculus in McCarthy and Hayes, 1969 considers instantaneous state of the world. Similarly, first-order temporal logic proposed by McDermott, 1982[70] was developed while considering partially-ordered, linear and dense structure of time. According to the logic proposed, time instants could be combined to form time intervals which could in turn be used to reason about non-zero duration events.

[70] *McDermott (1982). A Temporal Logic for Reasoning About Processes and Plans*

On the other hand, relative view of time is able to reason about imprecision and fuzziness in temporal information, since the order in which events occur is given priority to the exactness of when they occur. A notion of modeling time in relative terms was proposed in Kahn and Gorry, 1977[71], where the order of occurrence of events was taken into account to form *before/after* chains representing series of events. Extending relative view of time, Allen, 1983[72] noted that, given an event e, it can always be "magnified" and described at finer detail, which points out a shortcoming of absolute theory of time, since its time primitives (instants) are non-decomposable. Hence, the author proposed a temporal logic having time intervals as primitives and described possible relations between such intervals.

[71] *Kahn et al. (1977). Mechanizing Temporal Knowledge*

[72] *Allen (1983). Maintaining Knowledge About Temporal Intervals*

Time instants and intervals

It has been discussed in the literature that relying only on instant-based representation or only on interval-based representation does not lead to a comprehensive temporal logic. Therefore, these two representations were combined by some researchers. Vilain, 1982[73] proposed a temporal logic where relations between both time instants and intervals were described, however, both acted as primitives in

[73] *Vilain (1982). A System for Reasoning About Time*

the logic. A way of combining instants and intervals is by saying that intervals are bounded by non-decomposable instants which begin and end them, as described in Allen and Hayes, 1985[74], and can be given as pair of such instants.

[74] *Allen et al. (1985). A Common-sense Theory of Time*

Temporal entities

In the domain of temporal reasoning in AI, various temporal entities, describing occurrence of temporal propositions, have been proposed, which are compared in Vila, 1994[75]. For example, McDermott, 1982 defines *state* as the snapshot of world at a given time instant called its date. *Event* is defined as something happening and is associated to non-zero duration intervals. This is an extension to events being identified merely as state change. *Fact* is set of states over which it is true and may occupy an interval. Similar classification is proposed in Allen, 1984[76], where *properties* and *occurrences* are differentiated. *Property* defines static aspects of the world and is true over time. It is analogous to the definition of *Fact* considered by McDermott. *Occurrences* capture dynamic aspects of the world and are divided into *processes* and *events*. Here, *event* is defined as an activity having an anticipated outcome, whereas *process* is an activity without any culmination. Different authors consider *processes* and *events* in different ways and there is no agreed upon definition of these terms. Some definitions are listed in Galton, 2008[77]. Debate concerning definitions of *processes* and *events* is still a topic of discussion as it was raised during the workshop "Core Computations on Spatial Information" held during 10th International Conference on Geographic Information Science (GIScience 2018) and attended by the author of this thesis. Even though, the discussion remained inconclusive, some definitions for these terms were suggested: *Process* could be defined when there is change over some time interval and it is known what happened during that change, and *event* could be considered as instantaneous change in the state of the world. Even though these definitions may make sense to some researchers, they might not provide complete picture about change. As pointed out by Worboys, 2005[78], "One person's process is another's event, and vice versa."

[75] *Vila (1994). A Survey on Temporal Reasoning in Artificial Intelligence*

[76] *Allen (1984). Towards a general theory of action and time*

[77] *Galton (2008). Experience and History: Processes and their Relation to Events*

[78] *Worboys (2005). Event-oriented approaches to geographic phenomena*

Temporal structure

Even though, intuitively, time is considered to be *linear*, other ways of modeling time have been proposed in the literature. For example, a depiction of *branching* time is given in McDermott, 1982. The idea behind this model is that depending on events and/or actions happening at present, there could be multiple "possible futures", a notion which linear model of time cannot handle. In this representation, past is represented linearly (since it is relatively determinate) and future with multiple branches. Furthermore, *cyclic* representation of time, considered in Hornsby, Egenhofer, and Hayes, 1999[79], proves useful while reasoning about periodic or recurring phenomenon. For example, occurrence of tides and daily mobility patterns are periodic in nature, hence, require time to be cyclic for their modeling and analysis. Application of *linear*, *branching* and *cyclic* representation of time for visualising road traffic is described in Chen, Guo, and Wang, 2015[80].

79 *Hornsby et al. (1999). Modeling Cyclic Change*

80 *Chen et al. (2015). A Survey of Traffic Data Visualization*

2.2.2 *Spatio-temporal models*

Having clarified the notions of space and time, let us now see how they are integrated to describe and analyse spatio-temporal evolution of a phenomenon using spatio-temporal models.

Continuants vs. occurrents

Reality can be considered to consist of *continuants*, which persist through time, like objects and locations, and *occurrents*, like events, processes and actions, whose existence is time-dependent, i. e. they occur for some bounded time. Grenon and Smith, 2004[81] differentiate between continuants and occurrents and consider them to be compatible. In the following, we characterize the proposed models according to these criteria. We focus more on the models based on continuants, however, some models based on occurrents are also mentioned.

81 *Grenon et al. (2004). SNAP and SPAN: Towards Dynamic Spatial Ontology*

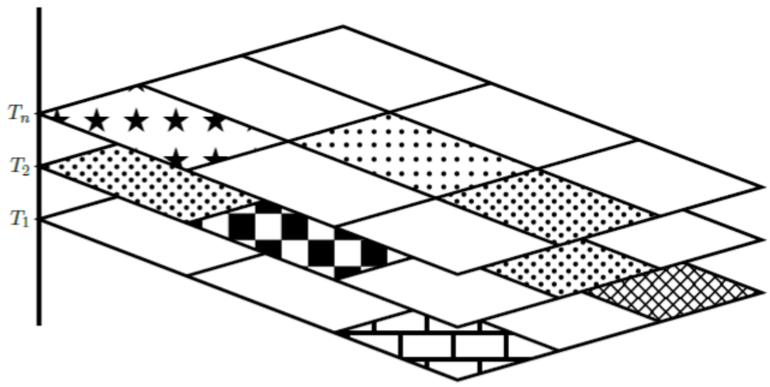


Figure 2.5: Layered structure formed using snapshot modeling approach. Figure is taken from Armstrong, 1988.

2.2.2.1 *Models based on locations and objects*

We first consider the category based on continuants - locations and objects. Differentiating between absolute and relative view of space, spatio-temporal models could be categorised as location-based and object-based. Siabato et al., 2018[82] considers similar categorisation while comparing different models proposed in the literature.

82 *Siabato et al. (2018). A Survey of Modelling Trends in Temporal GIS*

Snapshot model

One of the earlier works which describe change in attributes of a spatial location is proposed in Armstrong, 1988[83]. This model uses a snapshot view to describe spatial states at discrete time instants forming a layered structure (Figure 2.5). The comparison between two consecutive layers is required to understand the change in location attributes. Since the layered structure of spatial states leads to data redundancy, an approach associating attributes to individual locations and modeling their change with time is also considered.

[83] *Armstrong (1988). Temporality in spatial databases*

ST-composite model

In the same year, Langran and Chrisman, 1988 proposed a spatio-temporal model with the aim of avoiding data redundancy as in snapshot model described above, as well as for modeling change, from one state to the next, without requiring to compare consecutive states. This model, called Space-Time Composite, accumulates temporal changes on the initial state of the world using a polygon mesh. Every time a change occurs, new polygon representing the updated region is added to the initial state. However, since the new polygon has its own identity different from its temporal neighbours (i. e. polygons at previous time instant), issues regarding consistency of identity arise with this model.

Fuzzy set model

Another approach for modeling change at spatial locations with time is proposed in Dragicevic and Marceau, 2000[84] where fuzzy set theory is used to interpolate information between two consecutive snapshots of a given area. It uses the layered structure of states and generates new "fuzzy" layers between already existing layers to model spatio-temporal evolution between them (Figure 2.6). However, authors make a restrictive assumption that if no change is observed from one time-stamped layer to the next, no change occurred in the intermediate layers either. This may or may not be true.

[84] *Dragicevice al. (2000). A fuzzy set approach for modelling time in GIS*

Identifying entities

To model the evolution of spatio-temporal phenomenon over time, one needs to be able to identify distinct entities constituting that phenomenon. As pointed out in Yuan and Hornsby, 2008[85], in location-based models, the location helps in identifying entities so that their evolution can be analysed. In the models described above, the attributes at a location are compared at different time instants to see if any change occurred. On the other hand, object-oriented modeling approach facilitates the identification of individual objects using unique identities. Hornsby and Egenhofer, 2000[86] describes an

[85] *Yuan et al. (2008). Computation and Visualization for Understanding Dynamics in Geographic Domains*

[86] *Hornsby et al. (2000). Identity-based change*

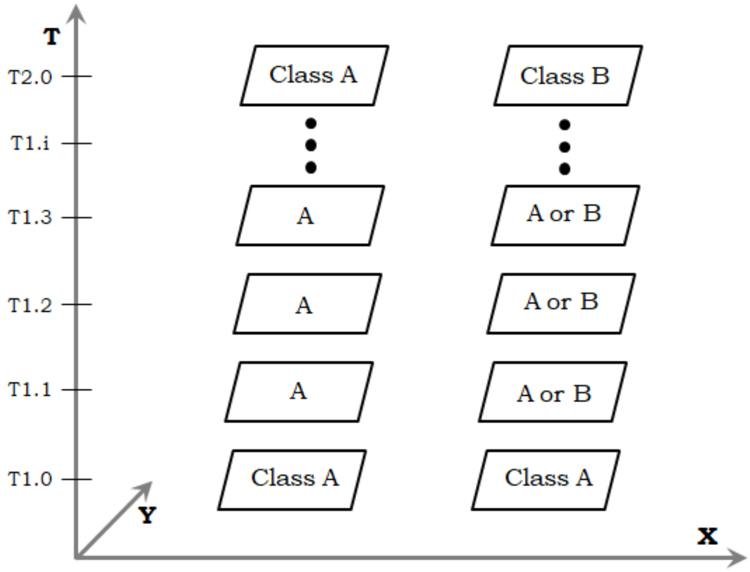


Figure 2.6: Two cases representing change in spatial locations. On left, no change occurred from time $T1.0$ to $T2.0$. On right, change occurred from class A to class B, with intermediate states having either class A or B. Figure is taken from Dragicevic and Marceau, 2000.

approach which employs object identity to track changes in object attributes.

Object-oriented model

Object oriented modeling provides a natural way to describe reality in the form of individual objects and supports classification, aggregation and inheritance, among other techniques, to describe model semantics. Egenhofer and Frank, 1989[87] introduced the idea of using object-oriented approach in the field of Geographic Information Systems as it was seen that existing relational models were insufficient to completely capture object-based reality.

[87] *Egenhofer et al. (1989). Object-oriented modeling in GIS*

ST-object model

Of the pioneer works on spatio-temporal object-based modeling, the one from Worboys, 1994[88] is noteworthy. It considers individual objects present in two-dimensional space and two-dimensional time (*valid* and *transational* time) and models them using spatio-bitemporal objects, consisting of spatial objects and bi-temporal elements. Spatial objects are represented using *simplexes* which represent their geometries and bi-temporal elements are represented using database and real-world time.

[88] *Worboys (1994). A Unified Model for Spatial and Temporal Information*

Object versioning model

Another model based on object-oriented approach is proposed by Wachowicz and Healey, 1994[89] which supports different versions of objects existing at different times (modeled as independent dimension). For each new object version, a different object identifier is generated and is stored with reference to initial state of that object. The generation of object versions is due to the occurrence of events, incorporated in the model, with events being associated to unique identifiers. Object versions are related to each other using a hierarchical structure representing past and future versions.

[89] Wachowicz et al. (1994). *Towards temporality in GIS*

Semantics behind change in objects

Change in objects deals with variation in shape, size, orientation and/or location of spatial primitives representing them. This variation results in changing spatial relations between objects over time. Egenhofer and Al-Taha, 1992[90] describe changes in topological relations between entities represented as regions in $\mathbb{R}^2$. Similarly, gradual changes occurring in orientation relations between spatial entities represented as 0-dimensional points are described in Freksa, 1992b.

[90] Egenhofer et al. (1992). *Reasoning about gradual changes of topological relationships*

Moving object model

Considering change in locations of objects results in objects being in motion, which are also modeled in the literature. For example, Moving Objects Spatio-Temporal data model (MOST) proposed in Prasad Sistla et al., 1997[91] abstracts moving objects in terms of their dynamic attributes which are time- and location-dependent and get updated as objects move. Such attributes use a motion vector (a time-dependent function describing object motion) and describe the state of the object at a given time.

[91] Sistla et al. (1997). *Modeling and querying moving objects*

Abstract data types for moving objects

There exists some research which models moving objects using different kind of spatial primitives. Depending on application requirements, a moving object can be modeled as a *moving point* or *moving region*. Erwig et al., 1999[92] proposed spatio-temporal abstract data types *mpoint* (moving point) and *mregion* (moving region) to facilitate such modeling. The value of type *mpoint* describes location as function of time and is represented as a curve in three-dimensions (2D space + time). The value of type *mregion* gives snapshot of the region at a given time and forms a volume in three-dimensions (2D space + time). The authors proposed operations on these data types which are able to model discrete and continuous variations in object locations and/or extents.

[92] Erwig et al. (1999). *Spatio-Temporal Data Types: An Approach to Modeling and Querying Moving Objects in Databases*

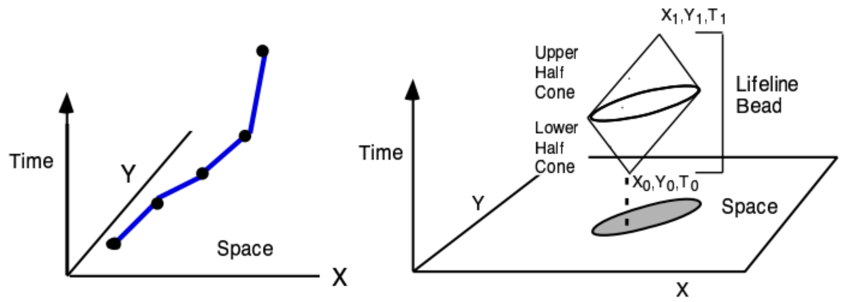


Figure 2.7: Geospatial lifeline and lifeline bead representing two ways of modeling moving point objects. Figure is taken from Hornsby and Egenhofer, 2002.

Geospatial lifeline and lifeline bead

Considering point-based approach with *mpoint* data type, the curve formed in three-dimensions (2D space + time) represents the trajectory of an individual or an object. Hornsby and Egenhofer, 2002[93] calls this trajectory a *geospatial lifeline* (Figure 2.7) representing the movement of an individual or an object over time. However, such geospatial lifelines determine a particular journey taken by the individual. For a more general representation, the concept of *lifeline bead*(Figure 2.7) is proposed which encompasses all possible locations in space-time where an individual can be present, given the start and end locations of the journey and travel speed. Furthermore, the author models the movement from different levels of detail, where finer levels reveal more information (like at what time the journey started, which places were visited during the journey, etc.) and coarser levels provide a general trend.

[93] Hornsby et al. (2002). *Modeling Moving Objects over Multiple Granularities*

2.2.2.2 *Models based on events and processes*

Even though Galton, 2008, while relating *events* and *processes*, argues that processes are not 'truly' occurrents, for the scope of this thesis, we put both events and processes in the same category, following the work of Grenon and Smith, 2004. Setting aside the debate behind definitions of events and processes, it can be agreed upon that both these temporal entities are required to fully comprehend and analyse spatio-temporal phenomenon. In this section, we will describe some models which apply event-based and process-based reasoning.

Event-based raster model

A pioneering work towards the development of event-based modeling approach was developed in Peuquet and Duan, 1995[94]. The proposed model, called Event-based Spatio-Temporal Data Model (ESTDM), stores the list of events which occur at distinct time instants over a set of locations. An event is defined as a 'significant' change in world

[94] Peuquet et al. (1995). *An event-based spatiotemporal data model (ESTDM) for temporal analysis of geographical data*

state. Starting from an initial state and its corresponding time, a time-stamped event list is maintained in chronological order of events, with each time stamp t_i associated to changes which occurred between t_i and t_{i-1}. Even though the model is suitable for sequential temporal queries, its implementation on raster data makes it restrictive.

Event-based vector model

Claramunt and Thériault, 1995[95] proposed a vector-based spatio-temporal model capable of formalizing different kinds of spatio-temporal processes between one or more entities over time. In this model, events are considered as set of processes which transform entities and processes are defined as aggregation of simultaneous or related changes. In a way, this model considers events to represent coarser detail than processes, however the lines between events and processes are blurred. The model differentiates between spatial, temporal and thematic domains and employs a versioning technique to store different versions of entities without redundancy.

[95] *Claramunt et al. (1995). Managing Time in GIS: An Event-Oriented Approach*

From static to event-oriented models

Worboys, 2005 advocates that occurrents (events and processes) should be considered as important as continuants (objects and locations) while modeling spatio-temporal phenomenon, and both object-oriented and event-oriented approaches are necessary for a comprehensive representation of reality. While the author does not differentiate between events and processes, he discusses four-staged development process of information systems, from static models to event-oriented models. In pure event-oriented algebraic model where, as author mentions 'everything is an event', time is represented as a collection of clock tick events and space is represented as a set of locations (conceptually similar to occurrents).

Event as set of processes

The model proposed by Yuan, 2001[96] uses a hierarchical structure to store data about events, processes, and states of a spatio-temporal phenomenon. It is developed to represent dynamic geographic phenomenon having both object and field properties in space-time. The model considers event as occurrence of something significant and process as a sequence of states demonstrating continuous changes in space-time. An event may relate to multiple processes and a process may relate to multiple states (Figure 2.8). Similar to Claramunt and Thériault, 1995, events represent coarser detail than processes and the model handles queries related to events and processes and their spatio-temporal relationships.

[96] *Yuan et al. (2001). Representing Complex Geographic Phenomena in GIS*

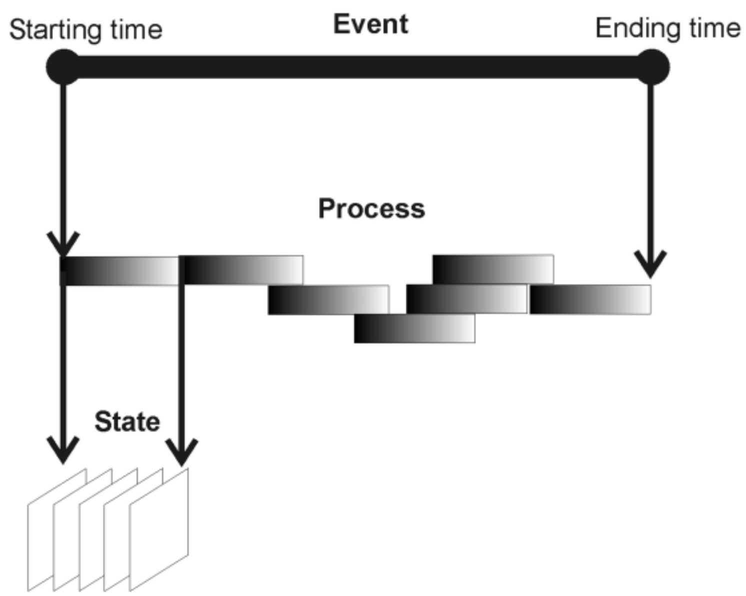


Figure 2.8: A conceptual structure of an event and its processes and states. An event is a spatio-temporal aggregate of processes, and a process is a sequential change of states in space and time. Figure is taken from Yuan, 2001.

The field of Spatio-Temporal Modeling deals with representing spatio-temporal phenomenon using a data model which takes into consideration various entities taking part in that phenomenon as well as their interactions. Two dimensions - Space and Time - along which the data models are defined play a significant role in determining model feasibility and complexity.
Generally, space is defined in terms of individual locations at which the phenomenon is observed (absolute point of view) or in terms of spatial relations between objects taking part in the observed phenomenon (relative point of view). These two standpoints for representing space change the meaning behind entities which are included in the data model. In case of absolute point of view, locations act as basic spatial entities whereas for relative point of view, objects acts as spatial entities. We consider relative point of view of space, hence, objects form basic spatial entities in our model.
Similar dichotomy is also considered for time. In case of absolute point of view, discrete time instants are required to model precisely the time at which an event occurs. For relative point of view, the order of occurrence of events is the main focus which is computed using relations between time intervals. However, some research indicates that both instants and intervals, when considered together, lead to more comprehensive models.

Hence, in our model, the considered time domain consists of both instants and intervals.
Furthermore, we consider two types of temporal entities, state and event, in our model. State represents the static snapshot of the model and event represents the change in state. Events can be continuous or discrete and both these point of views are considered in our model.
Considering objects as basic entities, the concepts from object-oriented modeling approaches are applied in our model. In such models, objects are represented using geometrical primitives like points, lines and polygons and spatial relations between geometries (objects) are explicitly defined according to the requirements. Various categories of spatial relations between static objects are: topological, orientation, direction, qualitative distance and linear order. In addition, for moving objects, relations such as relative trajectory and relative speed are also proposed in the literature.
Speaking of moving objects, some models specifically developed for moving objects are present in the literature. However, we have not based our model on one these models since they are more suitable for formalizing the movement of a single entity (represented using point or region). If multiple entities are to be modeled and their interactions (change in spatial relations between entities) are to observed, moving object models are not the correct choice.
In conclusion, the model proposed in this thesis is motivated from object-oriented models and considers continuants (objects) as well as occurrents (events) to model road traffic. The semantics behind included entities and their classification is discussed in the next chapter. In the next section, we discuss the characteristics of road traffic which make it a spatio-temporal phenomenon, hence, worthy of applying object-oriented modeling concepts to represent its spatio-temporal evolution.

2.3 ROAD TRAFFIC AS SPATIO-TEMPORAL PHENOMENON

Saying that road traffic is a spatio-temporal phenomenon leads to two questions: "Why is it a phenomenon?" and "Why is it spatio-temporal?". On the basis of the definition of *phenomenon* as proposed by Immanuel Kant, road traffic is a phenomenon since it can be observed to happen or occur, and data related to it can be collected using various kinds of sensors. It does not matter if it is modeled as a single entity or collection of individual objects, road traffic in general terms is a phenomenon since it is the "thing" which is under observation while developing traffic models.

Absolute point-of-view

The answer to the second question will be described in two parts: First, in case of empirical traffic models, like those discussed in Section 2.1, the data related to traffic is collected from different points in space and time using techniques like areal photography as discussed in Wardrop, 1952, powered car follower unit connecting lead and following car as used in Chandler, Herman, and Montroll, 1958 and induction-loop detectors as used in Treiber, Hennecke, and Helbing, 2000 and Maerivoet and Moor, 2005. This collected data is the observation about traffic (*theme*) collected at a given point in space (*location*) and time (*time*). Hence, according to Sinton, 1978, it has all the three components required to categorise it as spatio-temporal. Therefore, if an observation about a phenomenon is spatio-temporal, it is natural to assume that the phenomenon is also spatio-temporal. Moreover, the collected data is used to describe dependencies between traffic variables like *flow*, *density* and *speed* which are also location- and time-dependent.

Relative point-of-view

Secondly, if we take relative view to represent traffic, which is the case for the model proposed in this thesis, we need to take into account different objects which constitute traffic so that relations between them can be defined. Since traffic is an evolving phenomenon, some of whose constituents are under motion, the relations between them vary. In relative view, such relations are required for the mere definition of "space", and when they vary with time (due to the movement of traffic constituents) their evolution is spatio-temporal, since "space" (defined in terms of relations) varies with time.

Hence, from both absolute and relative perspectives, road traffic is a spatio-temporal phenomenon. Therefore, object-oriented modeling methodologies can be applied for its representation and analysis.

> In this section, we saw that road traffic is a spatio-temporal phenomenon, whether it is represented from absolute or relational point of views. Therefore, as mentioned previously, the concepts from the field of Spatio-Temporal Modeling can be applied to model traffic. To this end, we use relative point of view with object-oriented modeling approach.
> The characteristic of object-oriented models is that objects constituting the phenomenon, represented using geometrical primitives, act as basic spatial entities and spatial relations between object geometries represent their inter-dependencies. If the objective is to model such relations between various object geometries then object-oriented models discussed above can be

directly applied to model traffic. However, if one wants to focus on variations in spatial relations between objects, taking into account their geometry at every time instant leads to redundant information. Hence, we need to develop a model which is able to represent traffic from an abstract point of view.
In such a model, the type of object should determine which geometrical primitive is required to represent it. Once the geometry of each object is fixed, spatial relations between different objects can be defined. However, to model variations in these spatial relations over time, we do not need to consider geometrical primitives of the objects again.
A modeling methodology which is able to abstract the representation of the phenomenon while not excluding the desired object-oriented point of view is developed in the field of Graph Theory. Graphs provide a natural way to represent different entities and relations between two entities in an abstract way and have been applied to model spatio-temporal phenomenon in the past. They are suitable for our requirements since distinct traffic constituents and binary relations between them (it is noteworthy that we consider binary relations only) can be represented using nodes and edges of the graph. Furthermore, the evolution of traffic can be described by incorporating temporal dimension in the graph making it time-varying.
In the next section, we will present some related work where graphs have been applied to model spatial and spatio-temporal phenomenon.

2.4 GRAPHS FOR SPATIO-TEMPORAL MODELING

In relative view of the world, relations between distinct entities describe their inter-dependencies. Modeling spatio-temporal evolution constituting these entities requires that variations in their corresponding relations should also be considered. The structure produced by such entities and their corresponding relations provides an abstract view of the entire spatio-temporal phenomenon. In addition, its structural properties and their variation with space and time lead to a better understanding of the influence of these entities on each other and on the entire phenomenon being modeled.

Batty, 2003[97] calls for the incorporation of such relational concepts for spatio-temporal modeling of real-world phenomenon, and graphs provide a natural way to do so. They not only provide a formal model to represent static structures but they are also able to represent change. The author argues that the bottom-up approach, where local interactions between entities are abstracted to model the entire phenomenon

[97] *Batty (2003). Network Geography: Relations, Interactions, Scaling and Spatial Processes in GIS*

at global scale, provides better understanding about processes which lead to such changes.

Graphs for modeling static structures

Historically, graphs have been used to model (static) spatial structures for various applications. For example, Jiang, Claramunt, and Klarqvist, 2000[98] uses a connectivity graph for modeling spatial structure of a building, with nodes of the graph representing individual rooms or corridors and edges representing their adjacency. The authors utilise the concept of *space syntax* to model space from a cognitive point of view. It is argued that perception of large-scale space (space beyond human perception) is dependent on the perception of individual small-scale spaces (space within human perception). Such multiple small-scale spaces form nodes of connectivity graph which are linked together depending on some criteria, for example, adjacency, as mentioned before.

[98] *Jiang et al. (2000). Integration of space syntax into GIS for modelling urban spaces*

Another interesting and well studied application of graphs in representing spatial structures is in modeling street networks. In the literature, two ways to model streets using graphs have been proposed. The first, called *primal graph* approach, considers street intersections to be represented using nodes with edges representing the streets connecting two adjacent intersections, while the second approach, called *dual graph* representation, considers streets as nodes of the graph with edges connecting them if the corresponding streets intersect. Porta, Crucitti, and Latora, 2006[99] compare these approaches and discuss different centrality measures for both. While primal graph representation is more intuitive and is able to model spatial distance between two intersections, the dual graph is more suitable for analysing the topological structure of street networks without having spatial restrictions.

[99] *Porta et al. (2006). The Network Analysis of Urban Streets: A Primal Approach*

An application for modeling street networks using graphs is to describe the importance of various structural elements of such networks using graph properties. Claramunt and Winter, 2007[100] apply structural properties of street network graph for navigation and route guidance in urban environment. It is shown that incorporation of graph measures, representing salient features of a city, for navigation applications, leads to more intuitive route directions, since human cognition uses such features to model space.

[100] *Claramunt et al. (2007). Structural Salience of Elements of the City*

Furthermore, Domingo, Thibaud, and Claramunt, 2013[101] propose a graph-based model to represent the spatial structure which includes buildings, land parcels and road segments as nodes and topological relations between them are represented using edges. The model supports separate graphs for different structural elements as well as a single graph, called *complete graph*, consisting of all nodes and edges. The authors propose different graph metrics to provide a qualitative evaluation of the spatial structure considered.

[101] *Domingo et al. (2013). A Graph-based Model for the Representation of Land Spaces*

Graphs for modeling dynamic phenomenon

In addition to modeling static spatial structures, graph-based techniques have also been used to model dynamic phenomenon. For example, Spéry, Claramunt, and Libourel, 2001[102] describe a directed acyclic graph (DAG) model to represent spatial changes in land parcels at different time instants. They consider five types of elementary changes in shapes of land parcels over time. The nodes of the graph correspond to individual parcels and the edges represent the links between them at distinct time instants. The time dimension included in the model becomes explicit from the direction of edges linking nodes (parcels) at consecutive times. The model combines the snapshot approach for modeling spatio-temporal phenomenon with graph-based formalism.

[102] *Spéry et al. (2001). A Spatio-Temporal Model for the Manipulation of Lineage Metadata*

A formal model to represent time-varying data using graphs as underlying structure is described in Stell, 2003[103]. The evolution of distinct entities is formalized using a dynamic set over a certain time domain which may have linear or non-linear (branching) structure. A relation, called *support*, is considered which has different semantics depending on the application. General notion behind this relation is that an entity at a succeeding time instant is dependent (in some way) on an entity at preceding time instant. A major contribution of this model is its ability to represent entities from different levels of abstractions at different times. It is able to handle four different types of changes between entities over time.

[103] *Stell (2003). Granularity in change over time*

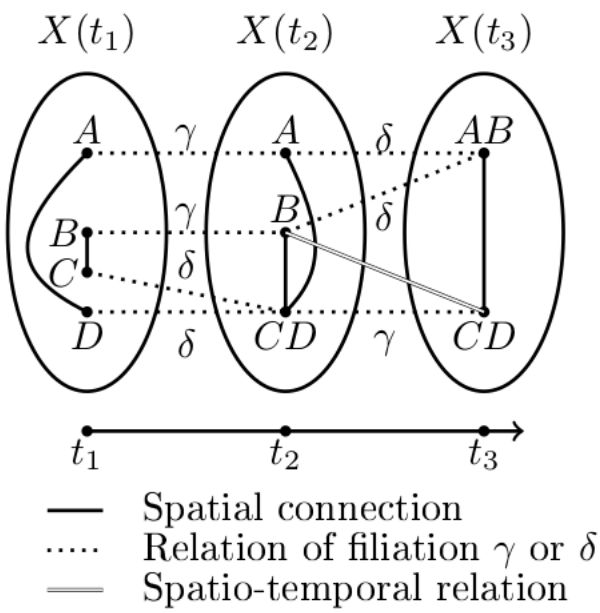


Figure 2.9: Spatio-temporal graph having spatial, spatio-temporal and filiation relations between entities existing at different time instants. The figure is taken from Del Mondo et al., 2010.

The model proposed in Stell, 2003 is extended in Del Mondo et al., 2010[104], where spatial and spatio-temporal relations are distinguished. Spatial relations are considered in entities present at one time instant whereas spatio-temporal relations exist between entities at different times. The authors also define a temporal *filiation* relation which is subsumed in the definition of *support* relation of Stell, 2003. Two kinds of filiation relations are considered: derivation (δ) and continuation (γ). Two entities A and B at times t_1 and t_2 respectively, have derivation relation if entity B, non-existent at time t_1, is derived from entity A at time t_2. The continuation relation between entities C and D at times t_3 and t_4 exists if C and D are essentially the same entity (but with different identities). Spatio-temporal graph consisting of these three types of relations between entities at different time instants t_1, t_2 and t_3 is shown in Figure 2.9.

[104] Del Mondo et al. (2010). *A Graph Model for Spatio-temporal Evolution*

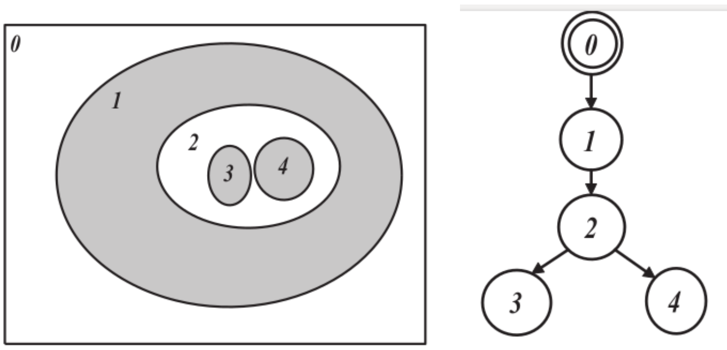


Figure 2.10: 2D objects represented using a rooted tree. Components 3 and 4 are surrounded by component 2, components 2, 3 and 4 are surrounded by component 1 and all these components are surrounded by component 0. Figure is taken from Jiang and Worboys, 2009

Combining event-based approach from Worboys, 2005 with graph-based formalism, Jiang and Worboys, 2009[105] propose a spatio-temporal model to describe change in topological structures of 2D objects, represented using rooted trees. Nodes of the tree represent the components which constitute an object with its edges representing *surrounded by* relation between two components (Figure 2.10). Snapshot of an object at a given time instant is said to undergo change over time when the structure of the corresponding tree changes. Various types of basic and complex changes in 2D objects are considered.

[105] Jiang et al. (2009). *Event-based topology for dynamic planar areal objects*

The models from Spéry, Claramunt, and Libourel, 2001, Del Mondo et al., 2010 and Jiang and Worboys, 2009 consider static graph structures at distinct time instants and represent the evolution in the modeled phenomenon by using temporal relations between distinct graphs. It is also possible to combine these graphs to form an aggregated graph structure and avoid information redundancy. This approach is followed by Costes et al., 2015[106], where graph snapshots are ag-

[106] Costes et al. (2015). *An aggregated graph to qualify historical spatial networks using temporal patterns detection*

gregated to form a spatio-temporal aggregated graph (Figure 2.11). However, in this case, the nodes of the aggregated graph are re-defined to represent the group of matched nodes or subgraphs of the graphs at individual time instants, while the edges are re-defined to represent the group of matched edges. Nodes and edges are said to be matched if they have spatio-temporal relations between them.

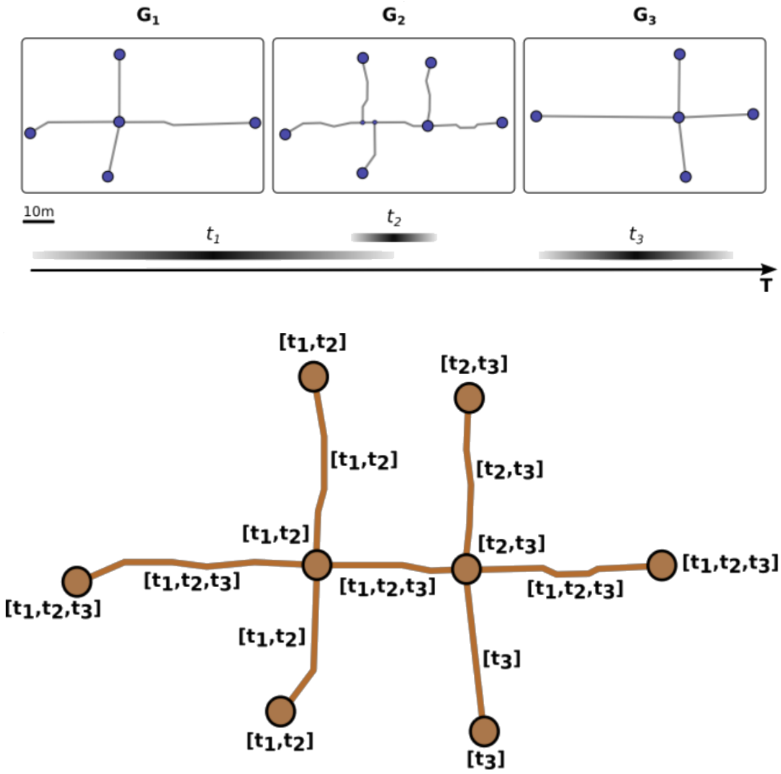


Figure 2.11: Graph snapshots combined to form spatio-temporal aggregated graph with re-defined nodes and edges marked by the time instants at which they are present. Figure is taken from Costes et al., 2015

From the above mentioned research, it is clear that graphs have been used for modeling spatial and spatio-temporal phenomenon in the literature. Static graphs are used to model spatial structures for applications like indoor navigation as in Jiang, Claramunt, and Klarqvist, 2000, route guidance as in Claramunt and Winter, 2007 or urban land space analysis as in Domingo, Thibaud, and Claramunt, 2013 and spatio-temporal graphs are used model changes in spatial relations between various entities of the modeled phenomenon as in Spéry, Claramunt, and Libourel, 2001 and Spéry, Claramunt, and Libourel, 2001. It is noteworthy that graph-based spatio-temporal models do not deal with representation of individual entities, rather they take a higher-level approach to represent various entities and corresponding relations.

Graphs for modeling road traffic

The reasons for applying graphs to model road traffic are twofold:

- As mentioned above, graphs will be able to represent traffic in terms of its constituents and their spatial relations from an abstract point of view, without taking into account the geometrical primitives, as is the case with object-oriented models.

- Using graphs, with nodes representing entities and edges representing relations, we will be able to model the underlying structure of traffic. This is an important advantage of using graphs, something which is missing in other object-oriented modeling methodologies. Having this underlying structure of traffic enables us to define *structure-based* traffic patterns which can be detected in the spatio-temporal graph representing traffic. This contribution of our work is discussed in Chapter 3.

To model a phenomenon from relative point of view, while taking into consideration various entities taking part in that phenomenon along with spatial relations between distinct entities, graphs are a natural modeling approach. Various graph-based models have been proposed in the literature to represent spatial and spatio-temporal phenomenon from an abstract standpoint. This abstraction facilitates the representation of change in relations between entities without considering the geometrical primitives required to model them. In addition, graphs are able to formalize the underlying structure formed due to the interactions between various entities. In case of road traffic, such an underlying structure can be applied to re-define the meaning of traffic patterns, as discussed in Chapter 3.

In the next chapter, we will formalize the graph model containing different entities which constitute road traffic. The included entities are classified into various classes, as is also done in various object-oriented models. To define relations between included entities, the geometrical primitives, using which they are represented, are also discussed. Then temporal dimension is incorporated into the model making the graph time-varying and adding the functionality to represent change in graph structure as well as change in node and edge attributes.

3

Spatio-Temporal Graph Model of Road Traffic

In the previous chapter, we developed the motivation to model road traffic in relative space-time and put forth the idea of using graphs to represent the structure formed by different interacting entities and resulting relations between them. Since road traffic evolves in space and time, it is important to merge these two dimensions while describing it. In this chapter, we will present a graph-based model of road traffic which captures its spatio-temporal nature by considering time-varying spatial relations between different entities. In addition, time-varying structure of the graph, resulting from spatio-temporal evolution of traffic, is taken into account.

We will start with describing the spatial graph (excluding time) to provide the reader with an understanding of different types of entities and spatial relations included in the model as well as different point of views of modeling road traffic. Then we will include the temporal dimension in the model, making it spatio-temporal.

3.1 Spatial graph model

Since the proposed model is based on graphs, formed using set of nodes and set of edges, the first question we ask is "What constitutes nodes and edges?". To answer this two-part question, let us first consider the case of nodes. Since we represent road traffic in relative space, the entities which take part in defining traffic as a phenomenon need to be taken into account, which form the nodes of the graph.

3.1.1 *Entities as graph nodes*

Given that entities constituting road traffic form the set of nodes, the next question we need to answer is "Which entities should be included in the model?". To find an answer to this question, we turn to the research conducted in the field of ITS, especially for the development of Advanced Driver Assistance Systems (ADAS).

In the last two decades, tremendous amount of research has been conducted for the development of ADAS with the goal of reducing traffic accidents and making vehicles safe. Research in this field is on the trajectory of developing autonomous vehicles which perceive their environment and, either aid the driver or make driving decisions

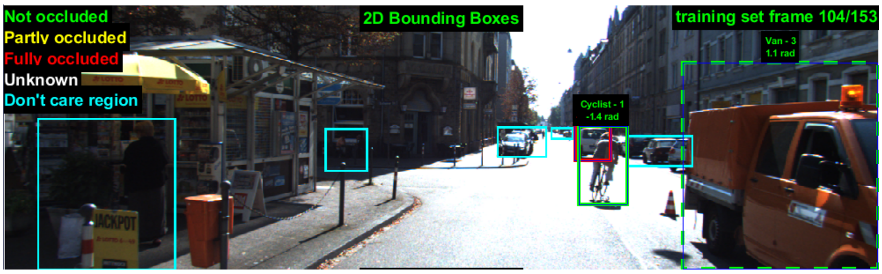


Figure 3.1: Image frame taken from KITTI Dataset showing 2D bounding boxes around different objects detected.

themselves, depending on the level of autonomy. Such a keen research and industrial interest has resulted in the development of several environment perception algorithms, surveyed by Zhu et al., 2017[107], based on vision sensors like camera, LiDAR and RADAR, mounted onboard vehicles or on intersections, buildings or road side infrastructure, which are able to detect static objects like buildings, traffic signs and traffic lights, road surface markings like lane boundaries, center lines, zebra crossings as well as track the movement of dynamic objects like vehicles, pedestrians and bicycles. These algorithms are represented using the **Extraction** module in the global workflow described in Chapter 1.

[107] *Zhu et al. (2017). Overview of Environment Perception for Intelligent Vehicles*

Without going into detail of how such algorithms work, our aim here is to present the reader with different kinds of entities which are detected by such algorithms and can be added as nodes of the graph. For the development of perception algorithms, large amount of data regarding surroundings of vehicles, moving on roads, is collected using state-of-the-art equipment. Such datasets can be used to replace the perception sensors, included as **Data Sources** module in the workflow, if they are not available. In addition, these datasets contain information about different object classes and their attributes. Hence, **Extraction** module could also be replaced if such datasets are used. Two of many famous perception datasets employed in the development of ADAS are KITTI and Cityscapes.

KITTI dataset, described in Geiger et al., 2013[108], contains data collected using sensors like cameras, LiDAR and GPS/IMU inertial navigation system, mounted on a test vehicle for the city of Karlsruhe, Germany. The data is collected from diverse driving conditions ranging from highways and countryside roads to city roads. The dataset contains ground-truth data pertaining to eight different object classes: *car*, *van*, *truck*, *pedestrian*, *person (sitting)*, *cyclist*, *tram* and *miscellaneous*. Even though the number of object classes considered is limited, the dataset is famous for development of many autonomous driving applications due to synchronised multi-sensorial information provided which aids in object detection and tracking. An image frame provided in KITTI dataset is shown in Figure 3.1.

[108] *Geiger et al. (2013). Vision meets robotics: The KITTI dataset*

Cityscapes dataset, on the other hand, proposed in Cordts et al., 2016[109], has semantic description of urban traffic scenes as its main objective. The data is collected using cameras mounted on a moving vehicle for 50 different cities and over varying weather conditions. Since traffic scene understanding is the main application for this dataset, it contains 8 groups of 30 different object classes, ranging from static objects like *road*, *sidewalk* and *roadside vegetation* to dynamic objects like *pedestrians*, *bicycles* and *vehicles* etc., which could be detected in a typical urban environment. The dataset consists of images whose pixels are classified on the basis of object category to which they belong, as shown in Figure 3.2. This is known as semantic segmentation of images.

[109] Cordts et al. (2016). *The Cityscapes Dataset for Semantic Urban Scene Understanding*

Figure 3.2: Image frame taken from Cityscapes Dataset showing image pixels classified according to object class to which they belong. The image corresponds to the city of Cologne, Germany.

In addition to datasets constructed using vehicle on-board sensors, several datasets have been proposed which contain data acquired from stationary sensors mounted on buildings or intersections. One such dataset called Ko-PER, is proposed in Strigel et al., 2014[110]. It contains data acquired using eight cameras and fourteen laser scanners, mounted on lamp posts and traffic lights, providing a view of an intersection, and is developed for dynamic multi-object detection and tracking applications. It takes into account four object classes: *cars*, *trucks*, *pedestrians* and *bikes* (Figure 3.3). In addition to laser scanner and camera data, reference data for vehicles using GPS-IMU is included.

[110] Strigel et al. (2014). *The Ko-PER intersection laserscanner and video dataset*

Taking motivation from environment perception algorithms and various datasets developed in the field of ITS, in this thesis, we consider nine different object classes into which nodes of the proposed graph model can be classified: *Vehicle*, *Building*, *Vertical Structure*, *Road Marking*, *Roadside*, *Road Segment*, *Pedestrian*, *Bicycle* and *Intersection*. The object classes considered include both static and dynamic objects. Different real-world objects included in classes with different types of objects are shown in Figure 3.4. Such objects form the set of nodes of the proposed graph.

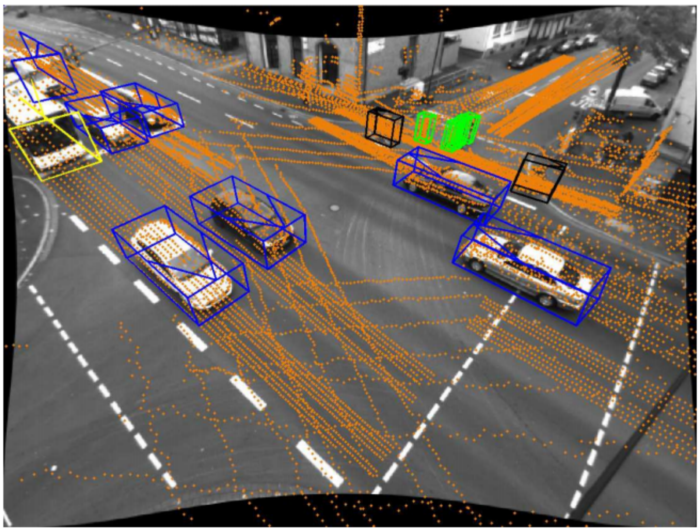


Figure 3.3: Image frame and laser scanner data provided by Ko-PER Dataset showing 3D bounding boxes for various object classes: blue boxes represent cars, black boxes represent bikes, yellow box represents truck and green boxes represent pedestrians.

3.1.2 *Spatial relations as graph edges*

Having described which real-world objects, classified into different classes, form the set of nodes of the proposed graph, let us now discuss what the edges of the graph will represent. Our objective is to model the interactions between different objects in relative space. These interactions give rise to spatial relations between objects. In the graph model, representation of such inter-object interactions is achieved by embedding them in graph edges, connecting nodes which represent the objects, with multiple spatial relations being represented using edge labels.

Since we consider different types of objects in our model, various kinds of spatial relations existing between these objects have to be considered to provide a comprehensive representation of road traffic. To model these relations, objects are represented using points, lines or polygons/regions. Relations considered between different object classes are listed in Table 3.1, with *T* being topological relations, *O* being orientation relations, *RT* representing relative trajectory, *RS* representing relative speed, *QD* being qualitative distance, *Ord* being (linear) order relations, *D* representing directional relations, *A* representing accessibility relation and *RO* representing road orientation relation.

3.1.2.1 *Vehicle-centric relations*

The elements of object class Vehicle have relations with elements of every other class since the proposed model is vehicle-centric and these

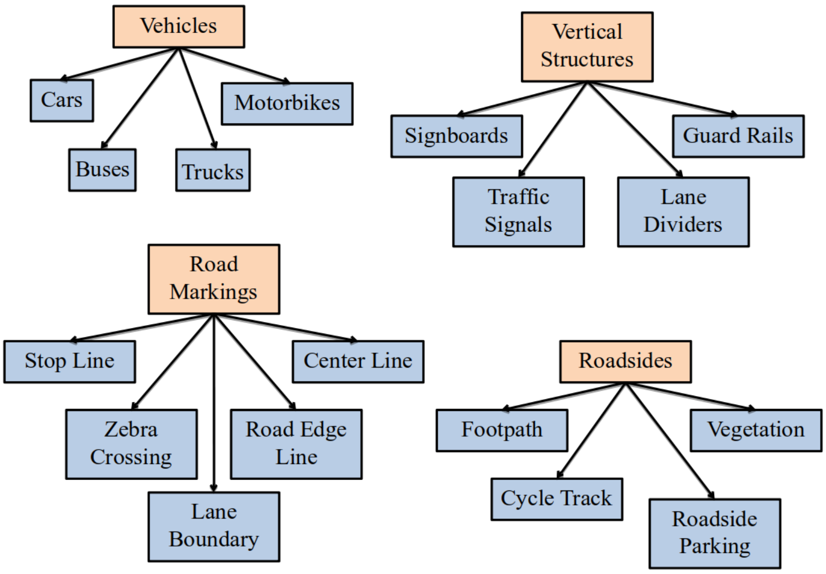


Figure 3.4: Different object classes and corresponding real-world objects considered in the proposed model.

Object class	Relation with class	Set of relations
Vehicle	Vehicle	$\{T, O, RT, RS, QD, Ord\}$
Vehicle	Building	$\{T, O, QD\}$
Vehicle	Vertical Structure	$\{T, O, QD\}$
Vehicle	Road Marking	$\{T\}$
Vehicle	Roadside	$\{T, QD\}$
Vehicle	Road Segment	$\{T\}$
Vehicle	Pedestrian	$\{T, O, RT, RS, QD\}$
Vehicle	Bicycle	$\{T, O, RT, RS, QD\}$
Vehicle	Intersection	$\{T, QD, Ord\}$
Pedestrian	Roadside	$\{T\}$
Bicycle	Roadside	$\{T\}$
Pedestrian	Road Segment	$\{T\}$
Bicycle	Road Segment	$\{T\}$
Pedestrian	Road Marking	$\{T\}$
Bicycle	Road Marking	$\{T\}$
Intersection	Intersection	$\{D\}$
Road Segment	Road Segment	$\{A, RO\}$

Table 3.1: Spatial relations between objects belonging to different classes.

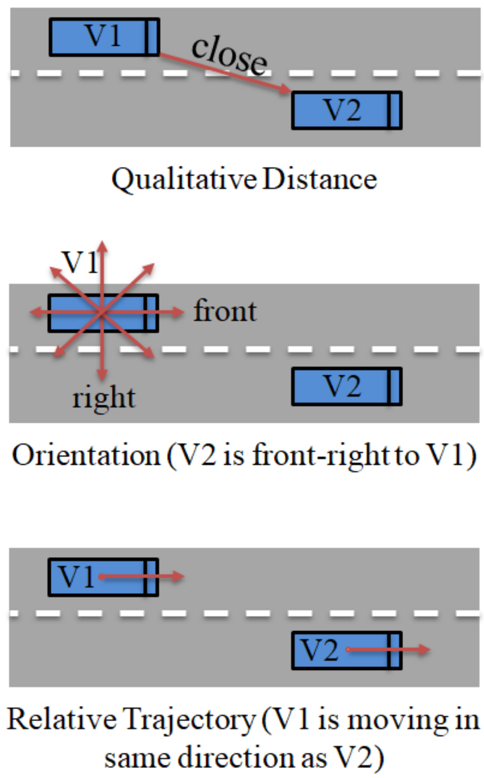


Figure 3.5: Some spatial relations between two vehicles. Vehicles are represented as regions or points and frame of reference is attached to the reference vehicle.

relations are necessary to model road traffic in relative terms, as vehicles are its primary elements. We consider six types of Vehicle-Vehicle relations. Topological relations (T) between vehicles are defined considering vehicle as a rectangular region representing its projection on a 2D surface. Using the formalism described in RCC8 from Randell, Cui, and Cohn, 1992[111], under normal traffic conditions, topological relation between two vehicles is disconnected (DC). However, if an accident occurs, this relation will change to a connected/overlap relation, with degree of overlap depending on the nature of the accident. Qualitative Distance (QD) relation describes distance between boundaries of vehicles and Order relation (Ord) describes the order in which vehicles move. Relative orientation (O) between vehicles is described by representing vehicles using points (like their centroids) and modeling where a point lies with respect to a reference point. To describe relative speed (RS) and relative trajectory (RT) of vehicles, direction vectors are associated to their centroids as is done by Renz, 2001[112]. Graphical representation of some of these relations is given in Figure 3.5.

[111] Randell et al. (1992). *A Spatial Logic Based on Regions and Connection*

[112] Renz (2001). *A Spatial Odyssey of the Interval Algebra: 1. Directed Intervals*

Between object classes Vehicle and Building, three spatial relations are considered. In general, topological relation (T) between vehicles and buildings, considering both as rectangular projections in 2D space,

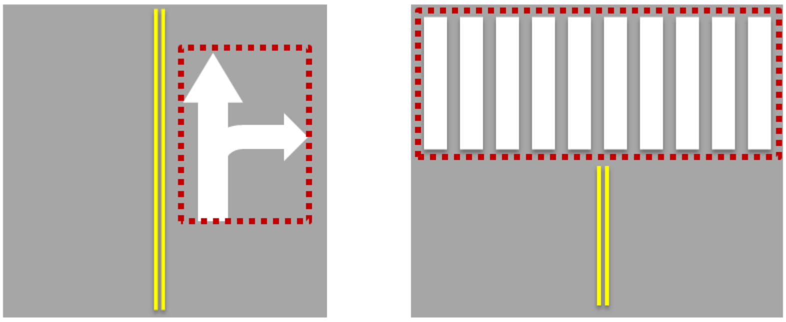


Figure 3.6: Road markings considered as 2D regions highlighted in red.

is disconnected (DC) but it could change in case of an accident. Qualitative Distance (*QD*) relation is defined between the boundaries of vehicles and buildings whereas orientation (*O*) relations require them to be represented using points.

These three relations are also described between classes Vehicle and Vertical Structure. However, the representation of vertical structures depends on their type. For example, sign boards and traffic lights could be represented using points or regions. In former case, topological relations (*T*) between them and vehicles are formalized using 9-IM model from Egenhofer and Herring, 1991[113]. In latter case, however, RCC8 is utilised. On the other hand, structures like guard rails and concrete lane dividers, which extend along the road segment, are represented using lines. In this case as well, topological relations are formalized using 9-IM model. For orientation relations (*O*), vehicles are represented using points and relative position of a point and a line is described. To formalize qualitative distance (*QD*) relation, vehicles are considered as rectangular projections and distance between their boundaries and vertical structures, represented as points or lines, are considered.

[113] *Egenhofer et al. (1991). Categorizing binary topological relations between regions, lines, and points in geographic databases*

Concerning object classes Vehicle and Road Marking, a single topological relation (*T*) is considered between them. However, different road markings are represented in different ways. For example, zebra crossings and direction arrows could be represented using 2D regions, as shown in Figure 3.6, whereas center lines, lane boundaries, stop lines at intersections and road edge lines could be represented using lines. The relation represents if the vehicle overlaps the road marking.

Two relations, topological (*T*) and qualitative distance (*QD*), are considered between objects of classes Vehicle and Roadside. For both relations, vehicles and roadsides are represented as 2D regions. Topological relation signifies if the vehicle is on the roadside with varying degree of overlap and Qualitative Distance relation describes the distance between vehicle and roadside boundaries.

Vehicle-Road Segment relation, which is a topological (*T*) relation, considers vehicles and road segments as 2D regions, with vehicles being tangential or non-tangential proper-part (TPP or NTPP) of road segments, if topological relation is modeled using RCC8. If multiple road lanes or bidirectional carriageways are considered, this relation could be specified for each lane or carriageway. The relation is TPP if the edge of the vehicle touches the edge of the road or the line representing lane boundary, and is NTPP otherwise.

Similar to Vehicle-Vehicle relation, relations between object classes Vehicle-Pedestrian and Vehicle-Bicycle are defined. Pedestrians and Bicycles are represented using either a point or a 2D region. Considering pedestrians/bicycles as points, topological relations (T) between vehicles and pedestrians/bicycles are defined using 9-IM model, and remaining relations (O, RT, RS, QD) are defined as mentioned above for other object classes. Considering pedestrians/bicycles as regions, RCC8 formalism is used to model topological relations.

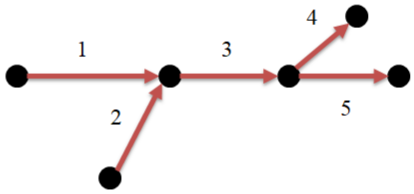
Representing intersections and vehicles as 2D regions, three types of Vehicle-Intersection relations are considered in the model. Topological (T) relations are described using RCC8 and Qualitative Distance (QD) relation describes the distance between the boundaries of vehicles and intersections. Order (Ord) relation describes if the intersection is in front or at the back of the vehicle. It is noteworthy that we do not include orientation relation between vehicles and intersections since vehicles move on road segments leading towards or away from intersections, resulting in intersections lying either in front of or behind the vehicles, which is captured by order relation.

3.1.2.2 *Other relations*

In addition to vehicle-centric relations, spatial relations considering other dynamic objects are also included in the model. For example, Pedestrian-Roadside, Pedestrian-Road Segment and Pedestrian-Road Marking relations, describe how pedestrians and roadsides, road segments and road markings respectively are related topologically. Here, topological relation (T) is considered to exist only if pedestrians are connected to roadsides, road segments or road markings, disconnected (DC) relation is not considered. Similarly Bicycle-Roadside, Bicycle-Road Segment and Bicycle-Road Marking relations are defined.

In the proposed model, object classes Intersection and Road Segment represent structural elements forming the road network. This road network is represented using primal and dual graphs described in Porta, Crucitti, and Latora, 2006[114] and discussed in the previous chapter. In primal graph, where nodes represent intersections, spatial relations between two intersections could be embedded within graph edges. We consider directional relations (D) between intersections, represented as points, describing their relative positions using cardinal directions. In addition, length of the road segment joining two intersections gives the metric distance between them, but it is not included as a relation since it is quantitative in nature.

[114] *Porta et al. (2006). The Network Analysis of Urban Streets: A Primal Approach*



	1	2	3	4	5
1	0	0	1	0	0
2	0	0	1	0	0
3	0	0	0	1	1
4	0	0	0	0	0
5	0	0	0	0	0

Figure 3.7: Accessibility between road segments described using adjacency matrix. Directed graph represents road network with nodes being intersections and edges being road segments. Adjacency matrix contains the information about accessible road segments. Figure is taken from Cheng, Haworth, and Wang, 2012

Similarly, edges of dual graph, with nodes as road segments, represent Road Segment-Road Segment relations. We consider two types of relations between road segments. Accessibility relation (A) describes if a road segment is accessible from another road segment. For this, edges between nodes representing these road segments must exist in the corresponding dual graph as well as traffic from first road segment should be allowed to move to the second road segment. This idea is described in Cheng, Haworth, and Wang, 2012[115] using adjacency matrices (shown in Figure 3.7). Road orientation (RO) relation describes relative orientation of a road segment with respect to another road segment. This is different from orientation relations described for other object classes since, in this case, road segments are not represented as points but as regions. The idea behind defining this relation is to explain the structure of an intersection at which road segments intersect. Choosing a reference road segment, the orientation of other road segments adjacent to it, at the given intersection, is described by this relation. In Figure 3.8, relation between road segments R_1 and R_2 describes that R_2 lies in part-3 with reference to R_1.

[115] *Cheng et al. (2012). Spatio-temporal autocorrelation of road network data*

3.1.2.3 *Frame of reference*

After elucidating different types of spatial relations we include in the proposed model, now we will take a look at how the frames of reference for these relations are considered. It is important to discuss frame of reference for every relation since different position or orientation of frame of reference changes the meaning behind the relation, as pointed out in Clementini, 2013[116] for orientation and direction relations. In this thesis, we will differentiate between *internal* and *external* frames of reference in 2D space. A frame of reference is internal if its position in inside the reference object, it is external otherwise.

[116] *Clementini (2013). Directional relations and frames of reference*

Since the proposed model is vehicle-centric, we first take a look at frames of reference for relations between class Vehicle and other classes. For describing topological (T), relative speed (RS), relative

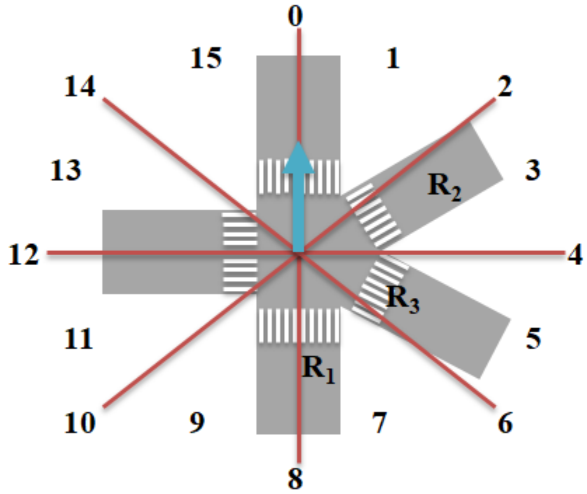


Figure 3.8: Structure of an intersection in terms of Road Orientation relation. Blue arrow represents the principle orientation of the reference road segment R_1 using which the frame of reference is defined.

trajectory (*RT*) and order (*Ord*) relations between Vehicle and other classes, external frame of reference with fixed position and orientation is considered. However, for orientation (*O*) and qualitative distance (*QD*) relations, internal frame of reference, having a reference vehicle-dependent position and orientation is considered, implying that its position and orientation changes with vehicle's motion. Relations between Pedestrian/Bicycle and various static elements (roadside, road segment and road marking) are described using external frame of reference. On the other hand, relations between elements of road network (Intersection-Intersection and Road Segment-Road Segment) are based on internal frame of reference whose position and orientation depends on the chosen reference intersection or road segment.

This notion of frame of reference is interesting for computing spatial relations using real-world data. If an internal frame of reference is required to describe the relation, it implies that, in case of dynamic objects, data from on-board perception sensors (camera or LiDAR) can be used to model relations between reference and target objects. For example, on-board LiDAR data provides information about the distance between a vehicle and a pedestrian which could be used to compute qualitative distance between them. Here, the LiDAR's frame of reference acts as frame of reference for computing the relation. Similarly, for external frame of reference, sensors mounted on buildings or intersections provide the data to compute spatial relations. For example, using a camera mounted on roadside infrastructure, order in movement of vehicles can be computed.

With the knowledge of different kinds of objects included as nodes in the graph and spatial relations between those objects embedded as graph edges, let us now dive into the formalisation of the graph itself and discuss how different point of views of modeling road traffic are embedded in the proposed spatial graph.

3.1.3 *Spatial graph*

Formally, a graph is given as a pair

$$G = (X,\ E) \tag{3.1}$$

where, X represents the set of nodes and $E \subseteq X \times X$ represents the set of edges connecting pairs of nodes present in set X. Both X and E are application dependent. In the graph model proposed in this thesis, set X of nodes consists of various real-world objects classified into nine different object classes: Vehicle (V), Building (B), Vertical Structure (VS), Road Marking (M), Roadside (F), Road Segment (R), Pedestrian (P), Bicycle (H) and Intersection (I), hence, is given as the union of sets of objects belonging to these classes. Mathematically,

$$X = V\ \cup\ B\ \cup\ VS\ \cup\ M\ \cup\ F\ \cup\ R\ \cup\ P\ \cup\ H\ \cup\ I \tag{3.2}$$

and can be considered to consist of all objects present in a geographical area (for example a city). However, since perception sensors play a significant role in detecting these objects, the number of objects included is limited since only those objects which are detected by these sensors, either mounted on-board vehicles or on infrastructural elements, are considered as nodes of the graph. Sensor data about such objects is utilized to compute relations between them.

Various types of spatial relations included between these objects are given in set

$$\Delta_B = \{T,\ O,\ RT,\ RS,\ QD,\ Ord\} \tag{3.3}$$

These relations are represented using graph edges. The set of edges is formalized as

$$\begin{aligned} &E = \{(x,\ y) \mid x\ \gamma_B\ y\},\ \gamma_B \in \Gamma_B,\ \Gamma_B \in \Delta_B, \\ &\quad \forall\ x,\ y \mid (x \in V\ \wedge\ y \in X)\ \vee\ (x \in P \cup H\ \wedge\ y \in F \cup R \cup M) \end{aligned} \tag{3.4}$$

Edges are considered to exist between nodes representing vehicles and any other class of objects, pedestrians and road segments, pedestrians and roadsides, pedestrians and road markings, bicycles and road segments, bicycles and roadsides and bicycles and road markings. Here, γ_B represents the value of the relation of type Γ_B between nodes x and y.

Depending on the class of nodes, set of spatial relations considered between them differs, as listed in Table 3.1. This graph G, referred to as the Basic Graph, contains most detailed and complete information about the road traffic elements for a given geographical area.

3.1.3.1 *Road graph & intersection graph*

Managing the amount of information in Basic Graph about countless objects could become tremendously difficult. A solution to this problem is to focus on objects linked to or present on a single road segment or an intersection. This is achieved by zooming in on a road segment or an intersection present in the area and included in the Basic Graph G. By doing this, level of detail of the information remains unchanged, as compared to Basic Graph, but the number of objects under consideration reduces. Such zoom operation is called Graphical Zoom in Frank and Timpf, 1994[117]. This leads to a reduced number of nodes and edges included in the graph and gives rise to a subgraph of the Basic Graph. If we zoom in on a road segment, the resultant subgraph is called Road Graph. Similarly, if we zoom in at an intersection, the resultant subgraph is called Intersection Graph. These subgraphs contain objects present on the road segment or intersection as nodes, and relations between them are embedded using edges. Similar to Basic Graph, perception algorithms and sensors limit the size of set of nodes of Road Graph and Intersection Graph since only the perceived objects are included.

[117] *Frank et al. (1994). Multiple representations for cartographic objects in a multi-scale tree - An intelligent graphical zoom*

Formally, for a given road segment, Road Graph G_R is given as

$$G_R = (X_R,\ E_R) \tag{3.5}$$

Set of nodes X_R is a subset of set X of nodes of Basic Graph. It does not include any intersections present in the Basic Graph and includes only the road segment for which Road Graph is being currently defined, among other object classes. Considering road segment $R_i \in R$ for which Road Graph G_{R_i} is defined, the set X_{R_i} of nodes would be

$$X_{R_i} \subset X \setminus (I \cup \{R_j \mid R_j \in R,\ j \neq i\}) \tag{3.6}$$

This definition of set of nodes of Road Graph implies that, in our model, intersections do not form part of road segments. In fact, we consider that road segment is defined as the part of the road network which connects intersections without including them (Figure 3.9). Set of edges of G_R includes edges present between nodes of G_R and is a subset of set E of edges of Basic Graph. Formally,

$$E_R \subseteq X_R \times X_R,\ E_R \subset E \tag{3.7}$$

Similarly, Intersection Graph for a k-th intersection $I_k \in I$ is given as

$$G_{I_k} = (X_{I_k},\ E_{I_k}) \tag{3.8}$$

with set of nodes being

$$X_{I_k} \subset X \setminus (R \cup \{I_j \mid I_j \in I,\ j \neq k\}) \tag{3.9}$$

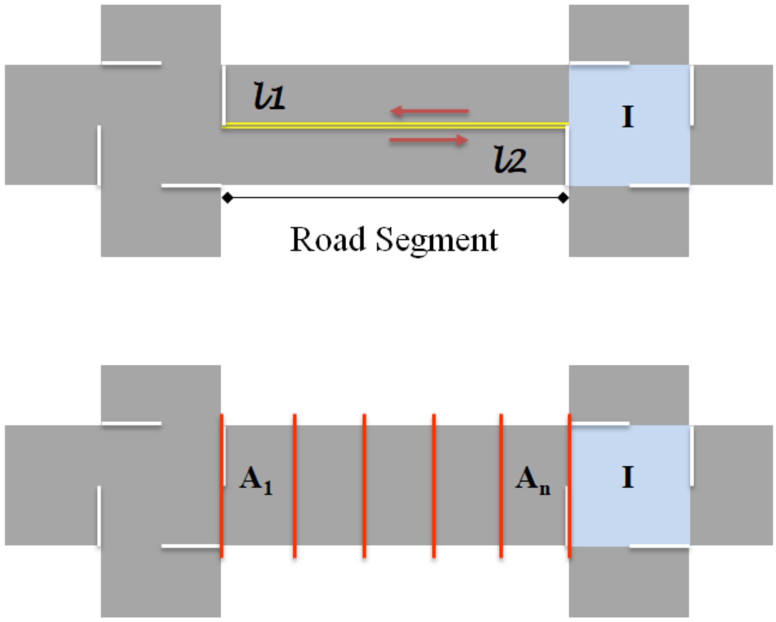


Figure 3.9: Road segment segregated in terms of longitudinal carriageways $l1$ and $l2$ and lateral sectors $A_1, ..., A_n$. Similar to road segment, intersection I is also represented using 2D region, highlighted in blue

including only the intersection under consideration with other object classes and excluding all road segments. Set of edges for this Intersection Graph is given as

$$E_{I_k} \subseteq X_{I_k} \times X_{I_k},\ E_{I_k} \subset E \tag{3.10}$$

Considering road segment as a 2D region, it forms the geographical area under consideration after zoom-in operation is performed. If the considered road segment is bidirectional, i. e. traffic flows in both directions, then two carriageways of this road segment can be differentiated, made explicit by the presence of concrete dividers or barriers (classified as Vertical Structure) or road center line (classified as Road Marking). These carriageways may be single or multi-lane and they segregate the road segment longitudinally. Another approach for segregating road segment is using lateral non-overlapping sub-parts or sectors, as employed in Kamran and Haas, 2007[118]. This approach takes into account the length of individual sectors or geometrical design of the road segment and does not depend on any physical barriers. These segregation approaches (Figure 3.9) result in different point of views for modeling road traffic.

[118] *Kamran et al. (2007). A Multilevel Traffic Incidents Detection Approach*

3.1.3.2 *Two-carriageway graph*

Using the former approach of longitudinal segregation of a road segment, less-detailed perspective of traffic can be computed. At this

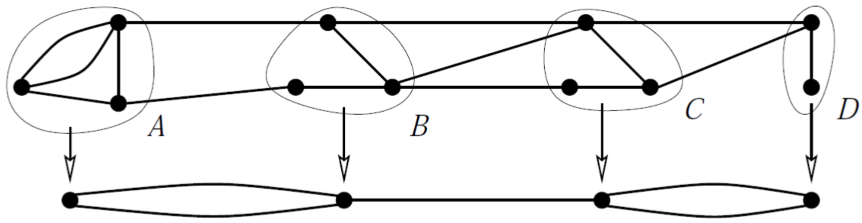


Figure 3.10: Granular subgraph with four nodes representing four subgraphs A, B, C and D whose nodes and edges are indistinguishable at coarser level. Figure is taken from Stell, 1999.

abstracted point of view, individual elements constituting traffic, moving in one direction, become indistinguishable and form an aggregated region of traffic over each carriageway. This region considers vehicles and other objects moving or present on a carriageway collectively, and is the result of performing Content Zoom-out operation described in Frank and Timpf, 1994. For a bidirectional road segment, two such regions are formed. In the Road Graph for the considered road segment, this abstraction is achieved by combining nodes, representing distinct traffic elements on each carriageway, to form a single abstracted node. This gives rise to a new granular subgraph (Figure 3.10), a concept described in Stell, 1999[119], of the Road Graph. This resulting granular subgraph is referred to as Two-Carriageway Graph.

[119] Stell (1999). *Granulation for Graphs*

The combination of different nodes of the Road Graph to form a node of Two-Carriageway Graph is performed on the basis of topological relation between a vehicle and a carriageway, i. e. if vehicle forms a tangential or non-tangential proper part of the carriageway (according to RCC8 formalism), corresponding vehicle node and its neighbouring nodes of Road Graph, are abstracted as node of Two-Carriageway Graph associated to that carriageway. Figure 3.11 shows Two-Carriageway Graph formed using Road Graph for a road segment $R3$. Objects shown in the figure belong to class Vehicle ($V1, ..., V5$), Building ($B1$ and $B2$), Vertical Structure ($S1$ and $S2$), Road Marking (M), Roadside (Fl and Fr) and Pedestrian ($P1$ and $P2$), and nodes representing group of vehicles present on two carriageways $l1$ and $l2$ are V_{l1} and V_{l2}, respectively.

Since the nodes of Two-Carriageway Graph provide coarser information about road traffic, spatial relations embedded in this graph are different from the ones described above. We consider four types of spatial relations: topological (T), average relative speed (ARS), directional (D) and relative traffic density (RTD), between two regions of traffic represented by the nodes of Two-Carriageway Graph. These relations are embedded in a single edge. Topological relation describes if two regions of traffic overlap. Under normal traffic conditions, there is no overlap and the relation remains disconnected (DC). However, if a vehicle moves to the wrong side of the road, either to overtake another

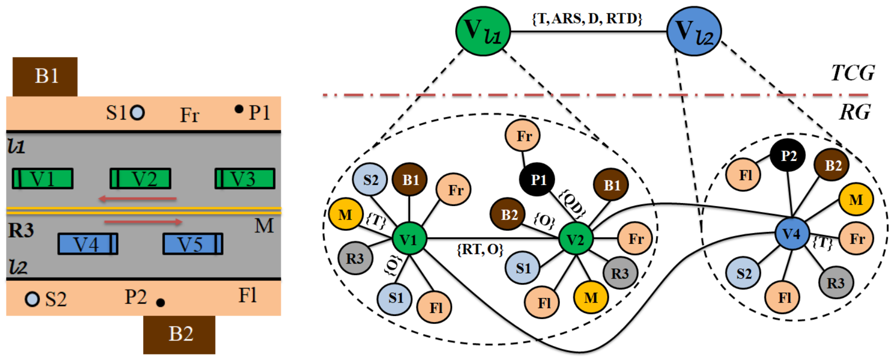


Figure 3.11: Abstracting Road Graph (RG) to form Two-Carriageway Graph (TCG) for road segment $R3$. Vehicles moving in opposite directions and their neighbouring objects are grouped together to form nodes of TCG. All nodes and relations are not shown to make the figure readable. Node color corresponds to color of object it represents.

vehicle or due to driver distraction, regions of traffic overlap, leading to a dangerous traffic situation. Average relative speed (ARS) relation describes if the traffic in one direction is moving faster or slower than in the other direction. Directional (D) relation describes the direction of movement of traffic using cardinal directions and Relative Traffic Density (RTD) compares the number of vehicles per unit length of the road segment on both carriageways.

Two-Carriageway Graph for a particular road segment is formalized as

$$G_{carr} = (X_{carr},\ E_{carr}) \tag{3.11}$$

Its set of nodes

$$X_{carr} = \{V_{l1},\ V_{l2}\} \tag{3.12}$$

consists of abstracted nodes V_{l1} and V_{l2}, combining distinct nodes for vehicles and neighbouring objects included in the Road Graph, for two opposite direction carriageways $l1$ and $l2$. Set E_{carr} contains one edge joining two nodes of G_{carr} and representing relations of types included in set $\Delta_{carr} = \{T,\ ARS,\ D,\ RTD\}$, between them.

3.1.3.3 *Sector graph*

The second approach for segregating the road segment results in another point of view to model traffic according to different representation scales. In this case, the road segment is divided laterally into smaller parts called sectors, with each sector containing subset of objects present on that road segment. The criteria for dividing the road segment into sectors depends on either the required length of the sectors or geometrical design of the road segment. With this approach, change in the size of a sector changes the representation scale of the model, while keeping the level of detail unchanged, as compared to

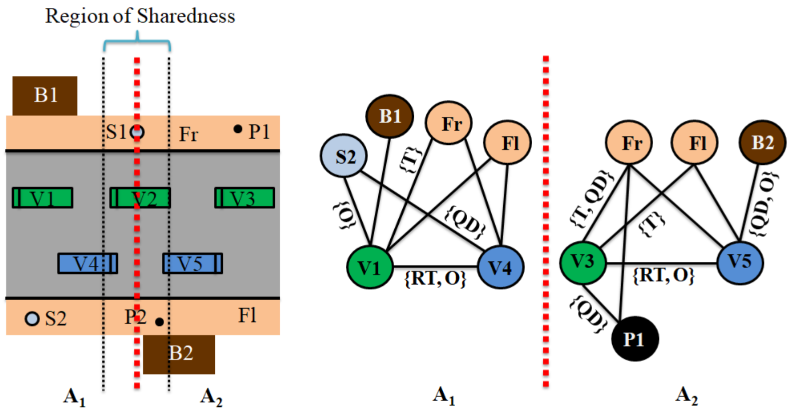


Figure 3.12: Road segment divided into two sectors A_1 and A_2 with Sector Graphs defined for both. Objects present in Region of Sharedness (RoS) are not included in any Sector Graph at given representation scale.

Road Graph. If the road segment is divided into smaller sectors, it provides more focused view of traffic as compared to larger sectors. Whatever the size, a Sector Graph is defined for each sector containing the objects present within it and the relations between them. Each Sector Graph is a subgraph of Road Graph for the road segment which is segregated (Figure 3.12).

The objective behind defining individual sectors is to reduce the number of objects considered as compared to the Road Graph. However, while defining individual sectors, some objects might be shared between adjacent sectors, leading to the dilemma whether to include them in corresponding Sector Graphs, and if yes, to which Sector Graph do they belong.

To solve this problem, first it is imperative to detect those objects which are shared between two adjacent sectors. For this, we propose the definition of Region of Sharedness (RoS), having fixed width, at sector boundaries, as shown in Figure 3.12, with sectors defined for a given representation scale. All objects, like vehicles, traffic signs etc., which are present within RoS are considered to be shared by adjacent sectors. However, objects, like footpaths, road edge lines, buildings etc. which are not completely included within RoS are not considered to be shared and are, therefore, included as nodes in Sector Graphs. As for the objects which are shared, if they are included in both Sector Graphs, it leads to the presence of redundant nodes. On the other hand, if we consider to include shared objects in one of the Sector Graphs, then it is not straightforward to decide in which graph they should be included. Hence, in our model, we do not include shared objects in any of the Sector Graphs at a given representation scale. However, if the representation scale is changed to form coarser (or larger) sectors, then previously shared objects might not be shared any longer between newly formed sectors and are, therefore, included in the corresponding Sector Graph, defined for coarser representation scale.

Given a Road Graph and a chosen representation scale for defining individual sectors, corresponding Sector Graphs are computed by taking subgraphs of Road Graph. These subgraphs are defined on the basis of geo-localization of individual nodes and maintaining existing edges between them. Here the representation scale varies from coarser (or global) to finer (or local) detail. The drawback of this approach is that if representation scale is varied, Sector Graphs have to be redefined. An approach for changing the representation scale from finer to coarser detail could also be defined. In this case, for a given set of sectors and their corresponding Sector Graphs, if two or more adjacent sectors are combined to form a coarser sector, their Sector Graphs have to combined by taking union of their sets of nodes and sets of edges, while also taking into account nodes corresponding to objects shared between adjacent sectors and forming new edges (if required) between nodes. This process of forming new edges makes this approach expensive.

Formally, Sector Graph for a sector of a road segment is given as

$$G_{sec} = (X_{sec},\ E_{sec}) \tag{3.13}$$

with its set of nodes $X_{sec} \subset X_R$ and set of edges $E_{sec} \subset E_R$ being subsets of sets X_R and E_R of Road Graph for that road segment.

3.1.3.4 *Primal and Dual Graphs*

As mentioned in the previous sub-section, structure of the road network is represented using primal and dual graphs, considering the set of intersections (I) or set of road segments (R) as nodes, respectively. Formally, Primal Graph is given as

$$G_P = (X_P,\ E_P) \tag{3.14}$$

Its set of nodes $X_P \subseteq I$ is the subset of set of intersections present in the considered geographical area and its set of edges represents direction (D) relations between intersections. Similarly, Dual Graph is formalized as

$$G_D = (X_D,\ E_D) \tag{3.15}$$

with $X_D \subseteq R$ being set of road segments and types of edges representing relations between them included in set $\Delta_D = \{A,\ RO\}$.

Figure 3.13 demonstrates the hierarchy of graphs proposed in this thesis. Since Basic Graph includes real-world objects as nodes and relations between them as edges, with the movement of dynamic objects, it will have to be updated regularly, which makes it a dynamic or time-varying graph. Other graphs derived from Basic Graph are also dynamic graphs. On the other hand, primal and dual graphs represent static structure of road network. Both these graphs enhance

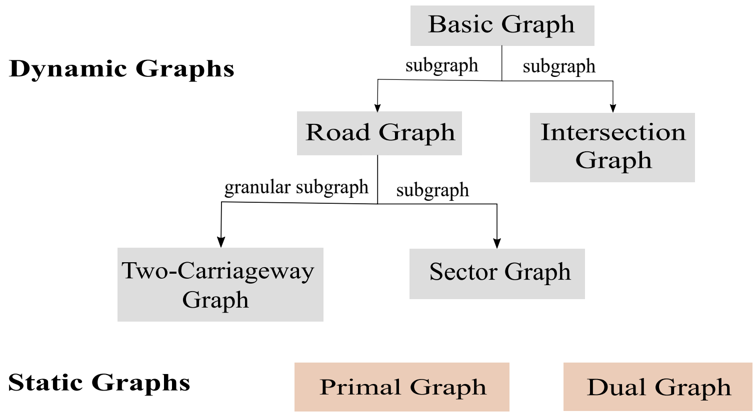


Figure 3.13: Hierarchy of proposed graphs including both dynamic and static graphs.

our model with extra information, and are related to Basic Graph since the nodes of both primal and dual graphs (intersections and road segments) are included as objects in Basic Graph.

Since Basic Graph and its derived graphs are dynamic, it is imperative that we consider temporal dimension in our model so that changes occurring in graph structure and in relations between objects, with movement of traffic over time, can be formalized.

To model road traffic in relative terms, its constituents and other elements which affect its flow, need to be identified. Furthermore, spatial relations between these elements need to be considered to describe their interactions in spatial terms. These elements and their corresponding relations produce an underlying structure which is modeled using a spatial graph, whose nodes correspond to these elements and edges connecting two nodes represent spatial relations between them.

We consider nine different classes into which traffic elements are classified by taking motivation from existing perception databases and research conducted for the development of self-driving cars. These object classes determine the type of spatial relations considered (Table 3.1).

The proposed spatial graph model describes interactions between objects from different point-of-views and leads to the definition of hierarchical structure describing inter-dependencies between them.

In addition to space, time needs to be considered implicitly in the model to incorporate variations in the graph since road traffic is dynamic phenomenon and time provides a dimension against which these dynamics can be described.

3.2 ADDING TEMPORAL DIMENSION

Time plays a critical role in modeling dynamic phenomenon. The spatial graph model described in the previous section represents a phenomenon (road traffic) which is inherently dynamic in nature since some of its constituents are under continuous motion resulting in variation in spatial relations between them and other static elements. Hence, the model describing such phenomenon is incomplete without the incorporation of temporal dimension.

Since the model proposed in this thesis is graph-based, including time in a graph model provides the functionality of formalizing variations in the graph, hence, making it time-varying or dynamic. Tremendous amount of research has been conducted in this field, so before describing our time-varying graph model, a brief state-of-the-art related to this field needs to be highlighted. The graph model which will represent road traffic should be able to handle highly dynamic phenomenon, i. e. those for which the structure of the graph is not known *a priori*. In addition, such a model should include elements (nodes and edges) which persist over time, since traffic constituents and their relations demonstrate such behaviour. Let us see if there exists a model with these characteristics.

3.2.1 *Time-varying graphs in literature*

The field of time-varying graphs is highly inter-disciplinary due to which many formalisms have been proposed in the literature which deal with and model graph dynamics. Our objective in this section is not to provide a comprehensive survey of the proposed approaches for modeling time-varying graphs, as in Holme and Saramäki, 2012[120]. Rather, we want to highlight some important contributions so as to describe the motivation behind the research conducted in this field and our own choice of time-varying graph model.

[120] *Holme et al. (2012). Temporal networks*

As pointed out in Harary and Gupta, 1997[121], time-varying graph models deal with variations in graph structure, i. e. nodes and edges being added or removed over time, and/or variations in their attributes over time. These variations can occur separately or simultaneously, in any combination, depending on the phenomenon being modeled. If they occur simultaneously, the graph is called *fully dynamic graph*.

[121] *Harary et al. (1997). Dynamic graph models*

One of the straightforward ways, extensively studied in the literature, of modeling time-varying graphs is to represent them using a set of static graphs for different time instants or intervals. The popularity of this approach is due to the fact that static graph theory is a well established field and by considering a time-varying graph in terms of static instantaneous snapshots, methods and algorithms developed for static graphs can be directly applied to time-varying graphs.

For example, Bhadra and Ferreira, 2002[122] describe a time-varying

[122] *Bhadra et al. (2002). Computing multicast trees in dynamic networks using evolving graphs*

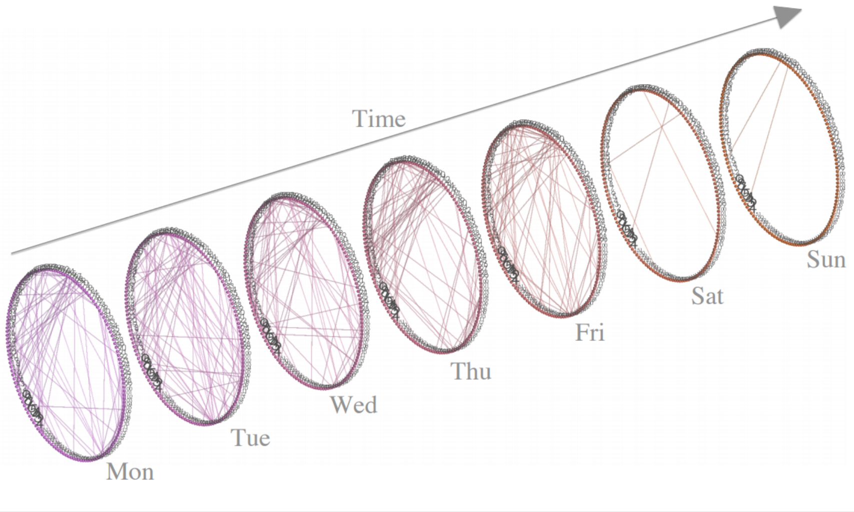


Figure 3.14: Sequence of static graphs representing weekly email interactions. Figure is taken from Tang et al., 2010

graph model, which they call evolving graph, for Fixed Scheduled Dynamic Networks (FSDN), given as set of static snapshots, with each snapshot representing network topology at a time instant. The model extends the notion of paths and strongly connected components developed in static graphs to time-varying graphs. Later in Xuan, Ferreira, and Jarry, 2003[123], path in the evolving graph is referred to as *journey* and defined as sequence of edges between nodes present at different time instants. The journey is said to be time-respecting when edges do not connect nodes from the past.

[123] *Xuan et al. (2003). Computing shortest, fastest, and foremost journeys in dynamic networks*

This notion of journey is also considered in Tang et al., 2010[124] where it is referred to as *temporal path*. Even in this case, the proposed model for time-varying graph, called *temporal graph*, is considered to be a set of static snapshots representing node interactions aggregated during a time window (Figure 3.14). Here, temporal path is used to derive temporal betweenness and temporal closeness centrality metrics which help in finding nodes playing central roles in the network. These metrics are temporal counterparts of static betweenness and centrality metrics employed in static networks.

[124] *Tang et al. (2010). Analysing Information Flows and Key Mediators Through Temporal Centrality Metrics*

Time-varying graph models described above consider static snapshots to be disconnected from each other. Even though for defining the notions of journey and temporal path individual snapshots need to be connected, this connection is not explicit in the model, but it is assumed that nodes of one snapshot store information over time, maintaining connectivity of the dynamic graph model. Moreover, if individual snapshots are kept disconnected, edges present at multiple time instants or during a time interval can not be modeled. This conceptual issue is partially solved by the model proposed in Basu et al., 2010[125] in which individual snapshots, called *graphlets*, are explicitly

[125] *Basu et al. (2010). Modeling and analysis of time-varying graphs*

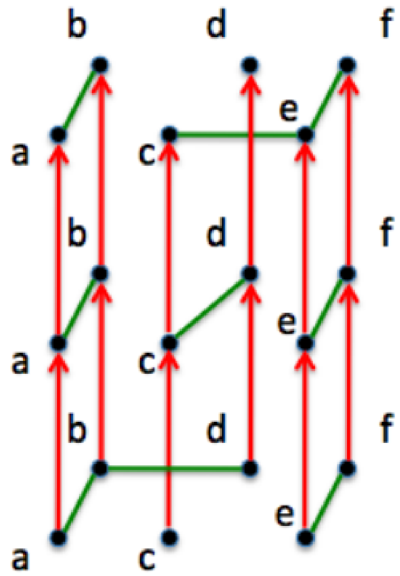


Figure 3.15: Instantaneous graph snapshots connected using temporal edges. Green edges connect nodes at same time instant whereas red edges connect same nodes at different time instants. Figure is taken from Basu et al., 2010

connected by edges between same nodes at different time instants (Figure 3.15). Using this approach, connectivity between nodes present at different time instants is described. If there exists a temporal path between nodes, they are said to be $\mathcal{T}$-reachable, where $\mathcal{T}$ is the time interval under consideration.

Continuing with the idea of connected snapshots, Wehmuth, Ziviani, and Fleury, 2015[126] describe a unifying model for time-varying graphs which distinguishes between four types of edges: Spatial edges between two nodes present at a given time instant, Temporal edges between same nodes present at distinct time instants, Mixed edges between distinct nodes present at distinct time instants and Spatio-temporal self loop edges connecting a node with itself at a given time instant. The unified nature of the proposed model is due to the fact that various types of models developed for time-varying graphs are in fact special cases of this model. Connecting individual snapshots with different kinds of edges leads to the possibility of defining spatio-temporal relations between nodes, as is done by Del Mondo et al., 2010[127].

Another approach for representing time-varying graphs is to attach time labels to the edges of the graph. These labels could represent the time at which the nodes communicate (in communication network) or the time at which a node infects its neighbour (to study spreading of a disease). Kempe, Kleinberg, and Kumar, 2002[128] present the notion of *time-respecting paths*, similar to that of journey from Xuan, Ferreira, and Jarry, 2003, in which time labels of consecutive

[126] Wehmuth et al. (2015). *A unifying model for representing time-varying graphs*

[127] Del Mondo et al. (2010). *A Graph Model for Spatio-temporal Evolution*

[128] Kempe et al. (2002). *Connectivity and Inference Problems for Temporal Networks*

edges are non-decreasing, and propose an algorithm for finding node-disjoint time-respecting paths in time-varying graphs. It is pointed out that by adding time labels on edges, traditional problems solved for static graphs become more complex. Similar approach is also used in Kossinets, Kleinberg, and Watts, 2008[129] to formalize the concept of *temporal distance*, i. e. minimum time required for information to spread from one node to another, in social networks.

[129] *Kossinets et al. (2008). The Structure of Information Pathways in a Social Communication Network*

Models based on snapshot approach, whether snapshots are disconnected, as in Tang et al., 2010, or are connected, as in Basu et al., 2010, represent dynamic phenomenon using sequence of static graphs. This modeling approach is suitable if the phenomenon being modeled is not highly dynamic, because if it is, such sequence of graph snapshots would be huge. Moreover, since this approach deals with aggregated snapshots, the dynamics occurring between two adjacent snapshots might get left out. Such problem can be solved by considering continuous time domain which is not the case for snapshot models described above. Similar argument can be given for models with time-stamped edges, as in Kempe, Kleinberg, and Kumar, 2002. Since these models deal with instantaneous interactions between nodes, they do not model edges present over time, which is a requirement for modeling road traffic using graphs.

Time-varying graph model proposed in this thesis is based on the model from Casteigts et al., 2012[130] which is able to describe variations in graph structure, and can be extended to model variations in node and edge attributes, over either discrete or continuous time domain. In this model, a time-varying graph is given as

[130] *Casteigts et al. (2012). Time-varying graphs and dynamic networks*

$$\mathcal{G} = (X,\ E,\ \mathcal{T},\ \rho,\ \varsigma) \tag{3.16}$$

where X is the set of nodes, E is the set of edges connecting these nodes, $\mathcal{T} \subseteq \mathbb{T}$ represents the lifetime of the system which forms a part of considered time domain $\mathbb{T}$, $\rho\ :\ E \times \mathcal{T} \to \{0,\ 1\}$ represents the edge presence function indicating if a given edge is present at a given time and $\varsigma\ :\ E \times \mathcal{T} \to \mathbb{T}$ represents latency function describing the time required to traverse a given edge at a given point in time. Since authors consider edge dynamics, the set of edges E varies with time and set of nodes X remains unchanged over time. Additional functionality could be added by considering time-varying node set. Similar to edge presence and latency functions, node presence and latency functions could also be defined. However, we consider edge traversal time to have zero duration, hence, we do not consider latency functions, for neither nodes nor edges, in our model.

The model from Casteigts et al., 2012 considers continuous time domain, which implies that graph structure can vary at any time during $\mathcal{T}$. This provides the user with the choice of discretizing the time domain and defining static graph snapshots either at every discrete time stamp $t_i \in \mathcal{T},\ i \in \mathbb{N}_{(>0)}$ ($\mathbb{N}_{(>0)}$ being the set of positive natural numbers) or only at those time stamps when the graph structure actually changes. Time stamps at which the graph structure changes are referred to as *characteristic dates* by Casteigts et al., 2012.

The reason to base our time-varying graph model on the one from Casteigts et al., 2012 is its the ability to handle both instantaneous edges as well as edges present over a time interval. In the context of road traffic graph we are proposing, this can be used to add an edge between two nodes, for example a vehicle and a pedestrian node, where a vehicle on-board sensor detects a pedestrian for an instant or over an extended period of time. In addition, the definition of edge presence function over the lifetime of the system provides an ambiguity on the time when an edge can be present, making this model suitable for highly dynamic networks where the structure of the network is not known *a priori*, which is the case for graph representing road traffic.

It needs to be pointed out that the model described in Wehmuth, Ziviani, and Fleury, 2015 is more general and can be used to represent the one from Casteigts et al., 2012 under certain conditions. However, we did not chose this model for our work since it utilizes Mixed edges, i. e. edges present between distinct nodes at distinct time instants, to bridge the gap with the model from Casteigts et al., 2012, and we do not consider such edges in our model. In addition, the graph model from Wehmuth, Ziviani, and Fleury, 2015 is developed for discrete time domain only.

Before describing the time-varying graph model in detail, let us first look at the structure of time and the types of temporal entities we consider in our model. Then we will add the temporal dimension to the spatial graph described in the previous section to make it spatio-temporal.

3.2.2 *Structure of time*

Vila, 1994[131] lists some characteristics of time described in the literature. Out of these characteristics, we consider time domain $\mathbb{T}$ to be linear, dense and positively unbounded $[0, \infty[$. We use notation $[\ ]$ to represent closed intervals and $]\ [$ to represent open intervals. Formally, $\mathbb{T} = \{t_1, t_2, ..., t_n\}$ represents the set of totally ordered time points with order relation $\leq$ between them. Here, the term 'time points' means points on continuous time line. Since the time domain is considered to be dense, its domain is the set of positive real numbers $\mathbb{R}_{(>0)}$ and $\exists t_j \in \mathbb{T} \mid t_i < t_j < t_k,\ \forall t_i,\ t_k \in \mathbb{T},\ i,\ j,\ k \in \mathbb{R}_{(>0)}$. It means that there exists a time point between any two given time points on continuous time line.

[131] Vila (1994). *A Survey on Temporal Reasoning in Artificial Intelligence*

Given two zero duration time points $t_{start} \in \mathbb{T}$ and $t_{end} \in \mathbb{T}$, a continuous interval bounded by them is given by $[t_{start}, t_{end}[$. This assumption is in accordance with the one described in Allen and

Hayes, 1985[132]. Similar to Casteigts et al., 2012, $\mathcal{T} \subset \mathbb{T}$ represents the lifetime of the model.

[132] Allen et al. (1985). A Common-sense Theory of Time

For the implementation of the model on a digital system, the time domain needs to be discretized according to a given temporal detail. We call individual points on discrete time line as 'time instants' which form the set of positive natural numbers with domain $\mathbb{N}_{(>0)}$. Similar to continuous intervals, discrete intervals are bounded by a pair of time instants and $\mathcal{T} \subset \mathbb{T}$ represents lifetime on discrete time line. Discretisation of the time domain makes it sparse, i. e. no time is considered to exist between two adjacent time instants.

3.2.3 *Temporal entities*

Taking motivation from McDermott, 1982[133], we consider two types of temporal entities in our model: *state* and *event*. State defines static aspects of the system. It represents the snapshot or condition of the system at a time instant or during a time interval (if the state persists over it). For example, state of road traffic at a given time gives information about traffic density at that time. Or, state of a moving vehicle during an interval gives the value of its speed (assuming it doesn't change during the considered interval).

[133] McDermott (1982). A Temporal Logic for Reasoning About Processes and Plans

Event, an occurrent according to Grenon and Smith, 2004[134], on the other hand, represents dynamic aspects of the system. It occurs at a *characteristic date* (the concept proposed in Casteigts et al., 2012) or during a *characteristic interval* and leads to state change. Events occurring at characteristic dates represent instantaneous events, which are either truly instantaneous (like deletion of edge in the graph) or are considered to be instantaneous due to lack of information about what happened during the event (like a vehicle hitting a pedestrian). If it is considered that an event occurs during some time interval, that interval is called characteristic interval and the event is considered to be continuous. For example, in a lane change maneuver, change from one lane to another could be considered instantaneous if information about the variation in steering wheel angle or change in vehicle speed during lane change is ignored. In this case, a vehicle is said to have changed its lane at a time instant t called the characteristic date of the lane change event. However, if the information about how the steering wheel angle, speed of the vehicle, its lateral and longitudinal acceleration etc. vary while performing lane change is considered, this event becomes continuous and the interval during which lane change occurs (from start to finish) is referred to as characteristic interval.

[134] Grenon et al. (2004). SNAP and SPAN: Towards Dynamic Spatial Ontology

3.2.4 *Spatio-temporal graph*

Having described the structure of time and temporal entities, now let us dive into the formalization of time-varying graph. We define the time-varying graph model as

$$\mathcal{G} = (X,\ E,\ \mathcal{T},\ \rho_X,\ \rho_E,\ \psi_X,\ \psi_E) \tag{3.17}$$

where, X and E are sets of nodes and edges, respectively, included in the Basic Graph described in the previous section, $\mathcal{T}$ is the system lifetime, $\rho_X : X \times \mathcal{T} \to \{0,\ 1\}$ is node presence function, $\rho_E : E \times \mathcal{T} \to \{0,\ 1\}$ is edge presence function, ψ_X is node labeling function and ψ_E is edge labeling function.

Theoretically, the time domain and, hence, the lifetime are considered to be continuous but can be discretized according to given temporal detail. Node and Edge presence functions in our model are taken from Casteigts et al., 2012. However, since the model from Casteigts et al., 2012 is a general model, the authors did not specify node and edge labels. In our model, both nodes and edges have attributes given as labels, hence, we define the labeling functions.

3.2.4.1 *Node labeling function*

In the previous section, we mentioned different object classes into which real-world objects were classified in our model. There are nine such object classes considered. In addition to these object classes, another class called 'Group', for the nodes of Two-Carriageway Graph which represent the group of vehicles moving in each direction, is considered for the sake of homogeneity. Hence, the total number of node classes now becomes ten and the set of these classes is given as $C_X = \{c_1,\ c_2,\ ...,\ c_{10}\}$.

The idea here is that a unique set of attributes can be associated to each node class and every node of a given class may have one or more class-specific attributes at a given time. For example, class Vehicle may have attributes like *Id*, *GPS location*, *Speed*, *Lateral acceleration* etc. and class Road Marking may have attributes like *Id*, *Location*, *Type* etc.

For an arbitrary node class $c_i \in C_X$, $1 \leq i \leq 10$, the set of associated attributes is given as $K_{c_i} = \{k_1,\ k_2,\ ...,\ k_m\}$ where the number of attributes m is class-dependent. For a given node $x \in X_{c_i}$, $1 \leq i \leq 10$, the value of its attributes is given by the attribute vector

$$\left[k_1(x),\ k_2(x),\ ...,k_m(x)\right] \tag{3.18}$$

This notation of representing attributes is taken from Zhou, Cheng, and Yu, 2009[135]. Here, the set $X_{c_i} \subset X$ is the set of nodes belonging to class $c_i \in C_X$. Such an attribute vector for the node $x \in X_{c_i}$ represents its label. Hence, the node labeling function, for a given node and its class, becomes

$$\psi_x(x,\ K_{c_i}) = \begin{bmatrix} k_1(x) & k_2(x) & ... & k_m(x) \end{bmatrix}_{1\times m} \tag{3.19}$$

[135] *Zhou et al. (2009). Graph Clustering Based on Structural/Attribute Similarities*

with $m = |K_{c_i}|$ being the cardinality of set K_{c_i}.

Considering all nodes belonging to class $c_i \in C_X$ and included in set X_{c_i}, the node labeling function can be extended as

$$\psi_{X_{c_i}}(X_{c_i},\ K_{c_i}) = \begin{bmatrix} k_1(x_1) & k_2(x_1) & ... & k_m(x_1) \\ k_1(x_2) & k_2(x_2) & ... & k_m(x_2) \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ k_1(x_n) & k_2(x_n) & ... & k_m(x_n) \end{bmatrix}_{n \times m} \tag{3.20}$$

where $n = |X_{c_i}|$ is the number of nodes belonging to class c_i. Note that functions ψ_x and $\psi_{X_{c_i}}$ do not include temporal dimension. Taking $\mathcal{T}$ as the lifetime of the system, time-varying attributes of the nodes belonging to a given class are given by a 3D vector of dimensions $|X_{c_i}| \times |K_{c_i}| \times |\mathcal{T}|$ where, $|\mathcal{T}|$ varies with the level of temporal detail considered. It gives the attribute vector for a given node $x \in X_{c_i}$ at a time instant t such that $\rho_X(x,\ t) = 1$ or during a time interval $[t,\ t'[$ such that $\forall t'' \in [t,\ t'[,\ \rho_X(x,\ t'') = 1$.

We do not restrict $\mathcal{T}$ to be continuous or discrete. But to be as general as possible, continuous lifetime can be considered which may be discretized into discrete time instants and/or discrete time intervals.

3.2.4.2 *Edge labeling function*

Similar to nodes, edges of the graph are also classified depending on the nodes they connect. For example, an edge between a vehicle and building node is classified as Vehicle-Building edge. Table 3.1 lists seventeen classes of edges connecting various object classes. In addition, the edge connecting the nodes of Two-Carriageway Graph is taken into account, making the total number of edge classes to be eighteen.

Given the set of node classes C_X, the set of edge classes is formalized as

$$C_E = \{c_p c_q \mid c_p, c_q \in C_X,\ 1 \leq p \leq 10,\ 1 \leq q \leq 10\} \tag{3.21}$$

where an element of C_E classifies an edge connecting nodes belonging to classes $c_p \in C_X$ and $c_q \in C_X$. It is possible to have an edge classified as $c_p c_p$ connecting nodes of classes Vehicle, Intersection, Road Segment and Group since edge classes Vehicle-Vehicle, Intersection-Intersection, Road Segment-Road Segment and Group-Group are allowed.

For an arbitrary class of edges $c_p c_q \in C_E,\ 1 \leq p \leq 10,\ 1 \leq q \leq 10$, its set of attributes is given as $R_{pq} = \{r_1,\ r_2,\ ...,\ r_v\}$ where the number of associated attributes v is class dependent. For an edge $e \in E_{c_p c_q}$, the attribute vector

$$\begin{bmatrix} r_1(e),\ r_2(e),\ ..., r_v(e) \end{bmatrix} \tag{3.22}$$

gives the value of its attributes and acts as its label. Here, $E_{c_p c_q} \subset E$ is the set of edges belonging to class $c_p c_q$. The edge labeling function for a given edge and its class is

$$\psi_e(e,\ R_{pq}) = \begin{bmatrix} r_1(e) & r_2(e) & \dots & r_v(e) \end{bmatrix}_{1\times v} \tag{3.23}$$

where $v = |R_{pq}|$. For all edges in set $E_{c_p c_q}$ belonging to class $c_p c_q \in C_E$, the edge labeling function becomes

$$\psi_{E_{c_p c_q}}(E_{c_p c_q},\ R_{pq}) = \begin{bmatrix} r_1(e_1) & r_2(e_1) & \dots & r_v(e_1) \\ r_1(e_2) & r_2(e_2) & \dots & r_v(e_2) \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ r_1(e_u) & r_2(e_u) & \dots & r_v(e_u) \end{bmatrix}_{u\times v} \tag{3.24}$$

where $u = |E_{c_p c_q}|$ is the total number of edges in set $E_{c_p c_q}$. Adding time dimension, edge labeling function for edges belonging to a given class are given by the 3D vector of dimensions $|E_{c_p c_q}| \times |R_{pq}| \times |\mathcal{T}|$. It gives the attribute vector for a given edge $e \in E_{c_p c_q}$ at a time instant t such that $\rho_E(e,\ t) = 1$ or during a time interval $[t,\ t'[$ such that $\forall t'' \in [t,\ t'[,\ \rho_E(e,\ t'') = 1$.

In our time-varying graph model, we have considered node/edge presence and labeling functions which are time-dependent, i. e. some nodes and/or edges may be added or removed from the graph as well as their attributes, represented as labels, may vary over time. This leads to two types of evolutions occurring in the graph.

3.2.5 *Graph evolution*

We refer to change in the structure or topology of the graph over time as *Topological Evolution* and change in the labels of nodes and edges as *Attribute Evolution* (Figure 3.16). The idea behind segregating the evolution of the graph into these two types is to be able to focus on each one separately. In addition, semantics behind temporal entities (state and event), described above, change depending on the type of evolution for which they are defined.

3.2.5.1 *Topological evolution*

In case of Topological Evolution, the objective is to model change in graph structure. For example, if a vehicle detects a pedestrian using an on-board perception sensor, an edge between nodes representing the vehicle and the pedestrian is added in the graph, which changes its structure. Or, if a vehicle, moving on a road segment $R1$, moves onto a road segment $R2$, after crossing the intersection connecting $R1$

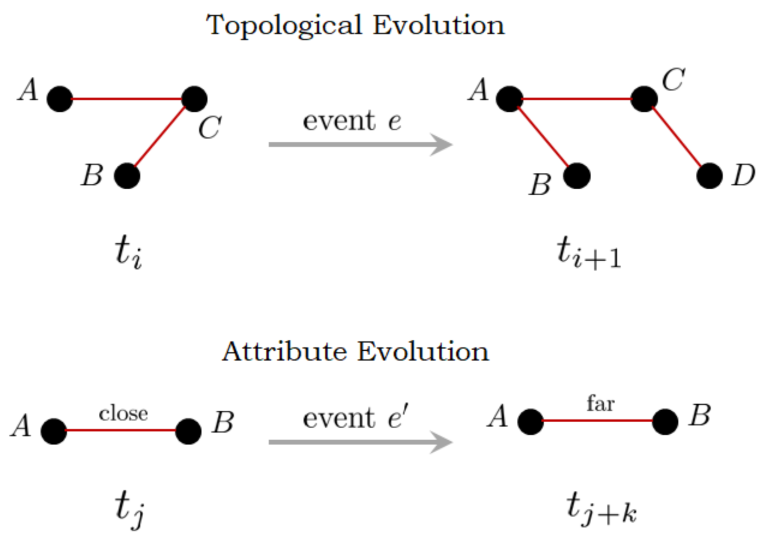


Figure 3.16: Two types of graph evolution. For Topological Evolution, topological event e is instantaneous and for Attribute Evolution, attribute event e' may or may not instantaneous. If it is instantaneous, $k = 1$ otherwise $k > 1$. For each time instant, topological and attribute state is shown.

and $R2$, it is included as a node in the Road Graph of $R2$ and removed from Road Graph corresponding to $R1$.

Since variation in node and edge labels is not the focus here, the formalization of the proposed time-varying graph model can be modified to

$$\mathcal{G}_T = (X,\ E,\ \mathcal{T},\ \rho_X,\ \rho_E) \tag{3.25}$$

considering only the functions related to structural elements of the graph.

We consider change in the graph structure, i. e. addition or removal of nodes and/or edges, to be instantaneous, which motivates us to consider the time domain to be discrete for this type of evolution. For every time instant $t \in \mathcal{T}$, a snapshot G_t of the time-varying graph $\mathcal{G}$ represents its *topological state* at time t. If a node or an edge is present over an interval, it implies that it is present at every time instant within that interval (considering discrete intervals) which can be modeled using presence functions ρ_X and ρ_E.

If at a given time instant, the structure of the graph changes, *topological event* is said to have occurred and that time instant is referred to as characteristic date. Now, it is possible that such topological events do not occur at every time instant in the considered lifetime. In this case, we could have same topological state at two adjacent time instants, leading to redundancy in the model. For this problem, Casteigts et al., 2012 propose to consider only the characteristics dates for defining graph snapshots.

3.2.5.2 *Attribute evolution*

The case of Attribute Evolution is different. Here, the objective is to model variations in attributes, given as labels, of nodes and edges. For example, the position of a vehicle, considered as an attribute of the node representing that vehicle, changes while it is in motion. Or, the orientation relation defined between two moving vehicles, given as attribute of the edge connecting these vehicles, varies over time.

This type of variation is described using labeling functions, hence, the formalization of the time-varying graph is modified to

$$\mathcal{G}_A = (X,\ E,\ \mathcal{T},\ \psi_X,\ \psi_E) \tag{3.26}$$

since the presence of nodes and edges is evident to define their respective labels.

For node and edge classes included in our model, various types of attributes can be defined. Some of these attributes, like moving vehicle's position or spatial relations between two moving vehicles, change frequently, while change in some other attributes, like traffic density of a road segment or spatial relation between a pedestrian and footpath, is less frequent. This also depends on the level of temporal detail at which the values of these attributes are acquired.

From a theoretical point of view, we consider continuous time domain to model such variations, which gives us the flexibility to adjust temporal detail according to the attribute being modeled. Such time domain is discretized at characteristic dates where *attribute state*, i. e. that value of the attribute, changes and *attribute event* is said to have occurred. In addition, to model continuous variation of some attributes, like position of a moving vehicle, characteristic intervals are defined during which attribute state keeps changing and attribute events keep occurring. In essence, characteristic intervals are counterparts of characteristic dates for continuous time domain.

As mentioned above, these two types of graph evolutions provide different perspectives to model variations in time-varying graph from graph-centric point-of-view, described in Casteigts et al., 2012. Both Topological and Attribute Evolution occur simultaneously making the proposed graph fully dynamic.

In addition to graph-centric point-of-view, Casteigts et al., 2012 consider node-centric and edge-centric point-of-views to describe graph evolution. Node-centric point-of-view is useful to formalize change, over time, in the neighbourhood of a node. For example, the objects detected, hence, the neighbourhood, of a moving vehicle vary, resulting in the sequence of neighbours of the corresponding vehicle node which could be used to model vehicle trajectory, given that the vehicle and other objects are geo-localized.

From an edge-centric point-of-view, graph evolution comes down to the presence or absence of a particular edge. Such is the case if an obstruction occurs between perception sensor and detected object, leading to the deletion of the corresponding edge in the graph. When the obstruction is cleared, the edge is added again.

Node and edge-centric point-of-views are not included in the scope of this thesis but they could provide additional functionalities to the proposed graph model. Two other concepts, described in Casteigts et al., 2012, for modeling the evolution of time-varying graph are that of *Underlying graph* and *Footprint*. Both these concepts are related to each other with a very subtle difference, which we explain in the following section, against the backdrop of the model proposed in this thesis.

3.2.5.3 *Underlying graph and footprint*

For a given time-varying graph (equation 3.17), Casteigts et al., 2012 define its underlying graph as $G = (X,\ E)$ to describe the *domain* of nodes and edges which (might) exist in $\mathcal{G}$. It is clarified in Casteigts, 2018[136] (Chapter 1) that underlying graph contains possible nodes and edges of $\mathcal{G}$, some of which may not appear in the considered time interval.

[136] Casteigts (2018). *Finding Structure in Dynamic Networks*

Footprint of $\mathcal{G}$, on the other hand, is the graph consisting of nodes and edges which appear at least once in $\mathcal{T}$. It is formalized as the union of individual graph snapshots occurring within $\mathcal{T}$ and defines the repository of nodes and edges appearing within $\mathcal{T}$.

It is noteworthy that the formalization of underlying graph and the Basic Graph (described in Section 3.1.3) is identical, leading to the belief that Basic Graph acts as the underlying graph of $\mathcal{G}$. This could be true in case when only the structure of the graph is under consideration. But when node and edge labels are also considered, the equality between Basic Graph and underlying graph becomes questionable. Basic Graph includes spatial relations, given as edge labels, between different nodes. Since underlying graph contains edges which may never appear in $\mathcal{T}$, there is no point in describing relations embedded on such edges. This functionality is not explicit in the formalization of Basic Graph. Similar to edges, attributes of nodes which never appear in $\mathcal{T}$ need not be considered.

This contradiction is solved by considering that all nodes and edges of Basic Graph appear at least once within $\mathcal{T}$ which makes it the footprint of $\mathcal{G}$ for interval $\mathcal{T}$. Footprint provides an aggregated perspective to graph evolution, which in terms of graph structure is given as union of graph snapshots for the considered time interval. In case of node and edges attributes, they can be aggregated over a time interval by considering their initial and final values at time instants binding that interval. Hence, in this case, we do not concern ourselves with how the attributes vary, rather which attributes vary, such that they have different initial and final values at interval-binding time instants.

In addition, variation in the size of the time interval for which the footprint of a time-varying graph is defined, provides additional functionality of changing level of detail at which graph evolution is considered. As Casteigts et al., 2012 note, coarser footprints (for larger intervals) describe general trends in evolution and finer footprints (for smaller intervals) describe frequent graph variations.

In this chapter we proposed a spatio-temporal graph to model road traffic. The model includes various objects which affect flow of traffic, as nodes, and its edges represent spatial relations between such objects. The proposed model incorporates the evolution of traffic over space-time, represented using variations in graph structure as well as node and edges attributes.

Variations in graph structure deal with addition or removal of nodes and/or edges which is considered to be instantaneous and occur at characteristic dates. On the other hand, variations in node and edge attributes might occur instantaneously or during some time interval (called characteristic interval), depending on the type of attributes and temporal resolution considered. These two types of variations lead to defining two perspectives to model graph evolution: Topological Evolution, dealing with change in graph structure and Attribute Evolution, dealing with change in node and edge attributes. Both these variations, occurring simultaneously, make the proposed graph fully-dynamic.

Since this spatio-temporal graph highlights the underlying structure of road traffic, in the next chapter, we will use this model to propose a novel point-of-view of formalizing traffic patterns which take into account structural elements of road traffic. We will take a look at various structure-based traffic patterns and propose an algorithm to detect such patterns in the spatio-temporal graph described in this chapter.

4

Structural Pattern Detection in Road Traffic

Previous chapter describes the formalization of a spatio-temporal graph model representing road traffic in relative terms by taking into account real-world objects, which affect the flow of traffic, and interactions between these objects, represented by spatial relations between them. The evolution of traffic is captured in the graph model in terms of variations in graph structure as well as in attributes of its nodes and edges. Since this model incorporates the constituents of traffic and their interactions, it unveils the fundamental structure forming the fabric of traffic for a given geographical area.

In this chapter, we will focus on this underlying structure and describe how it can be used for detecting spatio-temporal patterns in road traffic. We will start with presenting some research conducted for detecting traffic patterns, and highlight the difference between the definition of the term *pattern* in existing research and in the context of this thesis. Then we will take a look at some examples of traffic patterns which will provide the reader an understanding of the kind of patterns we consider in this thesis. Having done that, we will go on to present an algorithm to detect the said patterns in the spatio-temporal graph described in the previous chapter.

4.1 Traffic Pattern Detection

One of the research domains in the field of ITS, which has gained a lot of interest in the last few years, is Traffic Pattern Detection. In this section, we will highlight some research conducted in this field and accentuate the meaning behind the term *pattern* as defined in the related literature.

The process of acquiring meaningful traffic data has been a major part of research for traffic modeling using mathematical equations since 1950s. With technological evolution, data acquisition methods and data sources have changed. One of the famous data sources for traffic data is inductive loop detectors, embedded under road surface, which detect vehicles' presence over a certain part of the road segment. Using the data acquired with such loop detectors, Demiryurek et al., 2009[137] compute occupancy, volume and speed and use these parameters to model flow of traffic over a road segment. Here, *pattern* corresponds to the distribution of traffic flow over a certain time

[137] *Demiryurek et al. (2009). Towards Modeling the Traffic Data on Road Networks*

interval for a given road segment. Once variation of traffic flow is formalized, it is then used to cluster road segments of the considered road network according to their geographical locations. Inductive loop detectors are also employed by Jindal et al., 2013[138] to detect periodic variations in average vehicle speed for a given road network. The probabilistic distribution of average speed forms the *pattern* which the authors detect in the acquired traffic data. Another application for which traffic patterns are detected is to predict future traffic flow. Oh, Kim, and Hong, 2015[139] combine data from loop detectors with road network and weather data and propose an algorithm for traffic flow prediction. They define *pattern* as the feature vector which contains average speed, traffic volume, structure of road network and weather information and use probability density functions for traffic flow prediction.

[138] Jindal et al. (2013). *Spatiotemporal Periodical Pattern Mining in Traffic Data*

[139] Oh et al. (2015). *Urban Traffic Flow Prediction System Using a Multifactor Pattern Recognition Model*

In addition to non-perception data provided by loop detectors, perception sensors, like cameras, have been employed to monitor traffic and acquire traffic data. For example, Weiming Hu et al., 2006[140] use cameras mounted on roadsides to capture sequence of images in which vehicles are tracked over time using image processing algorithm. While tracking individual vehicles, their motion patterns are defined and if significant variation in these motion patterns is observed, it is considered as anomalous behaviour. The authors define motion *pattern* for a vehicle using probabilistic distribution function representing feature vector, describing vehicle trajectory, using its position and velocity information, computed in appropriate frame of reference.

[140] Hu et al. (2006). *A system for learning statistical motion patterns*

With vehicles getting equipped with GPS sensors and communication capabilities, another source for acquiring traffic data has come into picture. Communication capability of vehicles lead to the formation of Vehicular Ad-hoc Network (VANET) using which vehicles exchange information about traffic state around their current positions. Vaqar and Basir, 2009[141] employ this approach for estimating traffic state for a road segment and detect traffic congestion. Each vehicle communicates the position, speed and direction of its neighbouring vehicles. In this case, the feature vector combining this information acquired from every vehicle is considered as traffic *pattern*, describing traffic state, at a certain time instant.

[141] Vaqar et al. (2009). *Traffic pattern detection in a partially deployed vehicular Ad Hoc network of vehicles*

As can be seen from techniques described above, the term *pattern*, in the context of road traffic, is mainly associated to numerical description of traffic parameters combined to form feature vectors. This numerical description is the result of data acquisition techniques employed to collect traffic data. It will not be entirely untrue to say that traffic patterns which are detected are actually patterns in the acquired data about traffic. Such patterns are referred to as *statistical patterns* since statistical methods are used for their description and detection, and since these methods and related algorithms are mature and well-developed, such patterns are utilized in many different fields of research.

In contrast to statistical patterns which are based on numerical data, *structural patterns* are concerned with relationships between various constituents of an object or a phenomenon and are described using data structures reminiscent of underlying dependence between them. Generally, to describe structural patterns, *graphs*, *trees* and *strings* are used which, for detecting complex patterns, result in complex algorithms. Due to this fact, the field of structural pattern recognition is not as advanced as its statistical counterpart.

However, both statistical and structural patterns provide complementary information and a need for their unification has been felt in the past as noted in Bunke and Riesen, 2012[142]. In the context of road traffic, most pattern detection techniques are focused on detecting statistical patterns in traffic data but they do not take into account real-world objects which constitute road traffic nor their inter-relationships. In other words, they do not focus on the structure of traffic.

[142] *Bunke et al. (2012). Towards the unification of structural and statistical pattern recognition*

Since the graph model proposed in this thesis represents the structure of traffic in terms of its constituents and their inter-relationships, and graphs are preferred to represent structural patterns, in this chapter, we will discuss how graph-based pattern detection techniques can be applied to detect structural traffic patterns, in the proposed traffic model. Before diving into the details of the kinds of traffic patterns considered and the algorithm proposed, in the next section, we will take a look at some of the research which has been conducted in this domain.

> The research in the field of Traffic Pattern Detection relies on the acquisition of relevant traffic data which is carried out using different types of data sources. The focus of this domain is to detect patterns in the acquired traffic data. Hence, most data collected is numerical in nature for which statistical methods are developed for pattern detection. The detected patterns are referred to as *statistical patterns*.
> Researchers from Pattern Recognition community have also proposed another kind of patterns, called *structural patterns*, which describe relations between various constituents of an object or a phenomenon. Since both statistical and structural patterns provide complementary information, none of them can be completely discarded.
> Existing research in Traffic Pattern Detection focuses on statistical patterns. However, such traffic patterns do not exploit the underlying structure of traffic. In this chapter, we will describe the notion of structural traffic patterns which will provide a novel perspective to the related research.

4.2 GRAPH-BASED PATTERN DETECTION

In the domain of pattern detection and recognition, graph-based methods have been in use since 1970s. Vento, 2015[143], while providing a survey of the related literature, differentiates between three main periods of time - pure, impure and extreme - during which graph-based pattern recognition methods have been developed, although with different motivations. Pure period is so called since during this period, most of problems and algorithms proposed dealt directly with graphs. During impure period, the gap between statistical and structural pattern recognition was narrowed as the algorithms developed for statistical pattern detection were applied on graphs, and during extreme period, graphs are transformed into feature vectors so as to use statistical methods for structural pattern detection. For the scope of this thesis, we limit ourselves to pure approaches, i. e. those which deal directly with graphs for pattern recognition.

[143] Vento (2015). *A long trip in the charming world of graphs for Pattern Recognition*

When using graphs to represent patterns, an application deals with comparing two patterns and finding if they are similar or exactly the same. This is achieved by comparing corresponding graphs, and results in *exact* and *inexact* graph matching techniques. In case of exact graph matching, strict correspondence between two graphs is required while in case of inexact matching, the distance (graph edit distance) between two graphs is computed to verify their similarity. A survey highlighting both these approaches is given in Conte et al., 2004[144]. In this thesis, we only consider the case of exact graph matching.

[144] Conte et al. (2004). *Thirty Years Of Graph Matching In Pattern Recognition*

To find exact correspondence between two graphs, their nodes must be mapped while preserving edge adjacency (non-induced version) as well as non-adjacency (induced version). If such a mapping exists, the graphs are referred to as *isomorphic*. Checking if a graph is isomorphic to a subgraph of another graph is the problem of *subgraph isomorphism*, which is known to be NP-Complete, as noted in Cook, 1971[145]. Other versions of this problem like *subgraph monomorphism* and *subgraph homomorphism* are described in Conte et al., 2004, but in this chapter, we deal with the problem of subgraph isomorphism only.

[145] Cook (1971). *The Complexity of Theorem-proving Procedures*

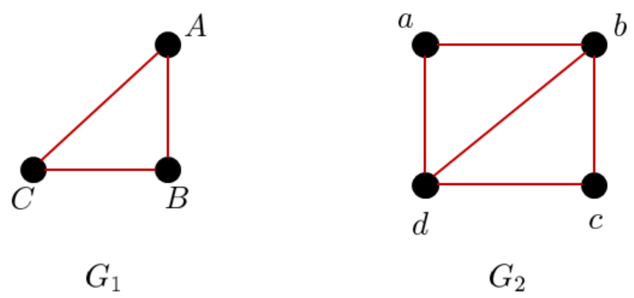
Formally, given two graphs $G_1 = (X_1, E_1)$ and $G_2 = (X_2, E_2)$, where G_1 is called *pattern graph* and G_2 is called *target graph* (Figure 4.1), with $\mu_{12} \subset X_1 \times X_2$ representing the mapping between them, G_1 is isomorphic to a subgraph of G_2 if

$$\forall x \in X_1,\ \exists x' \in X_2 \mid (x,\ x') \in \mu_{12} \tag{4.1}$$

$$\forall (x,y) \in E_1,\ \exists (x',y') \in E_2 \mid (x,\ x') \in \mu_{12},\ (y,\ y') \in \mu_{12} \tag{4.2}$$

$$\exists (x',y') \in E_2 \implies \exists (x,y) \in E_1 \mid (x,\ x') \in \mu_{12},\ (y,\ y') \in \mu_{12} \tag{4.3}$$

First condition checks for mapping of nodes of both graphs and second and third conditions check for node adjacency as well as non-adjacency, respectively, i. e. graph G_1 is isomorphic to an induced subgraph of graph G_2.



$$\mu_{12} = \{(A, c),\ (B, d),\ (C, b)\}$$

Figure 4.1: Subgraph isomorphism between pattern graph G_1 and target graph G_2 with one of many possible mappings μ_{12}

Subgraph isomorphism has been studied in case of both static as well as dynamic graphs. In the following sections, we will describe some related work for both types of graphs.

4.2.1 *Static graphs*

The problem of subgraph isomorphism has been widely studied in case of static graphs. Even though it is NP-Complete, some heuristics, such as node and edge labels in case of labeled graphs, and pruning strategies, for reducing possible mappings, are used to reduce computational time. Conte et al., 2004 describe various algorithms which have been proposed in the past for subgraph isomorphism.

Of the famous methodologies, tree-based state space search along with backtracking, proposed in Ullmann, 1976[146] is noteworthy. In this case, a tree representing the state-space with nodes corresponding to states i. e. partial mappings, is used to find potential mappings using depth-first search. If a mapping does not satisfy the isomorphism conditions, the algorithm backtracks to find another partial mapping and continue. The authors applied this technique for graph and subgraph isomorphism. Recently, Carletti et al., 2018[147] improved tree-based state space search with backtracking by considering node labels of both graphs and initial ordering of nodes of pattern graph. In addition, they pruned the search space by considering k-look-ahead rules to verify if a given partial mapping will lead to consistent mappings in next k steps.

[146] *Ullmann (1976). An Algorithm for Subgraph Isomorphism*

[147] *Carletti et al. (2018). Challenging the Time Complexity of Exact Subgraph Isomorphism for Huge and Dense Graphs with VF3*

In the literature, subgraph isomorphism has also been modeled as Constraint Satisfaction Problem (CSP). In this case, the nodes of pattern graph are assigned a *domain* containing nodes of target graph. Here, the objective is to reduce the size of the domain for each node of the pattern graph and map it to a node of the target graph while satisfying isomorphism conditions. Solnon, 2010[148] describes an algorithm for solving subgraph isomorphism modeled as CSP.

[148] *Solnon (2010). AllDifferent-based filtering for subgraph isomorphism*

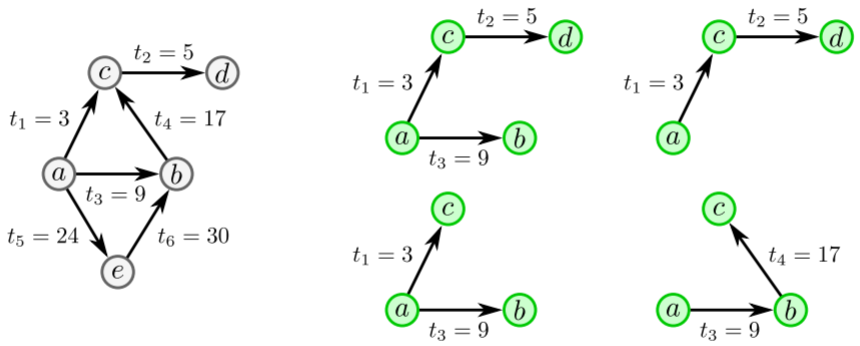


Figure 4.2: Isomorphic temporal subgraphs with $\Delta t = 10$ for target graph shown on the left. Figure is taken from Kovanen et al., 2011

Lastly, algorithms based on graph indexing are also employed for subgraph isomorphism. Such algorithms mainly focus on the decision version of the problem of subgraph isomorphism i. e. on detecting if subgraph isomorphism exists between a given pattern graph and a set of target graphs. They do not focus on finding where and how many times a subgraph isomorphic to the given pattern graph exists but if there exits one. An algorithm which employs path-based indexing technique to prune search space is proposed in Giugno and Shasha, 2002[149].

[149] *Giugno et al. (2002). GraphGrep: A fast and universal method for querying graphs*

4.2.2 *Dynamic graphs*

In case of dynamic graphs, the problem of subgraph isomorphism is extended to include temporal correspondence between pattern and target graphs. Kovanen et al., 2011[150] introduce the framework of temporal motifs, defined as isomorphic temporal subgraphs of dynamic graphs (Figure 4.2) whose edges represent temporal events, and propose an algorithm to find Δt-connected motifs, where two events are considered Δt-adjacent if time difference between their occurrence is Δt. This approach takes into account the time difference Δt while performing subgraph isomorphism.

[150] *Kovanen et al. (2011). Temporal motifs in time-dependent networks*

Since graph structure as well as temporal information embedded in it are required to find subgraphs isomorphic to a given pattern graph, they can be considered in different order. Redmond and Cunningham, 2016[151] propose three orders - Topology before Time, Time before Topology, and Time and Topology Together. In case of Topology before Time, temporal information in dynamic pattern and target graphs is ignored at first and graphs are mapped according to their structure. Then the temporal information is included to find those graphs which comply with it. In case of Time before Topology, the order is reversed and in the third case, they are considered together. The authors conclude that hybrid approach gives the best results.

[151] *Redmond et al. (2016). Subgraph Isomorphism in Temporal Networks*

The need to apply subgraph search in dynamic graphs also appears in case of streaming graphs, where stream of time-varying graphs is queried continuously to look for anomalies or unexpected behaviours. Instead of applying subgraph isomorphism every time the dynamic graphs gets updated, these algorithms perform graph matching in an online fashion while keeping track of partial mappings obtained in previous time steps. One such algorithm is proposed in Fan et al., 2011[152].

[152] Fan et al. (2011). *Incremental Graph Pattern Matching*

The algorithms from Kovanen et al., 2011 and Fan et al., 2011 do not suit our needs since they are proposed for dynamic flow networks and streaming graphs, respectively, which we do not deal with, in the context of road traffic model proposed in this thesis. However, we apply a similar perspective as Redmond and Cunningham, 2016 to consider graph structure and temporal ordering together while performing subgraph isomorphism search.

In the context of road traffic, the algorithm which we will propose in this chapter, applies an existing algorithm, called VF3 proposed in Carletti et al., 2018, which uses the technique of tree-based state space search with backtracking for solving subgraph isomorphism. We apply this algorithm, originally developed for static graphs, to the case of dynamic graphs. We base our choice on VF3 because it has been shown to be computationally efficient for large and dense graphs, it finds all solutions i. e. all mappings between a pattern and target graph pair and is shown to be memory efficient. This algorithm has been compared, using experimental evaluation, to other tree-based state space search as well as CSP based algorithms in Carletti et al., 2018[153].

[153] Carletti et al. (2018). *Comparing performance of graph matching algorithms on huge graphs*

Before diving into the algorithm description, we will discuss some examples of structural patterns in road traffic which can be detected using the spatio-temporal graph model proposed in previous chapter. For now, to describe the examples as well as the pattern detection algorithm, we consider various classes of nodes present in the model as well as time-varying structure of the graph, however, the type and value of spatial relations between different constituents of road traffic are not included. We are trying to answer the question: "What does the time-varying structure of graph, representing road traffic, reveal about the movement of dynamic objects?"

> Representing structural patterns using graphs has, as an application, the comparison of two patterns to check their similarity by comparing their corresponding graphs. If strict correspondence between elements of both graphs is sought, it leads to *exact* graph matching between both graphs. One form of exact graph matching is the problem of *subgraph isomorphism* where given two graphs, G_1 (called pattern graph) and G_2 (called target graph), a subgraph of G_2 isomorphic to graph G_1 is to be

detected. This approach is utilized to detect structural traffic patterns in this chapter.
The problem of subgraph isomorphism has been widely studied in case of static as well as dynamic graphs. The approach proposed in this chapter relies on a tree-based state space search algorithm called VF3, proposed in Carletti et al., 2018, which is developed for static graphs. We extend this algorithm to the case of dynamic graphs. The choice of VF3 is made on the basis of its performance with large and dense graphs and its ability to find all mappings between a given pair of pattern and target graph.

4.3 STRUCTURAL PATTERNS IN ROAD TRAFFIC

Since spatio-temporal graph model described in the previous chapter represents the underlying structure of road traffic, let us now discuss how it can be applied in the context of traffic pattern detection by presenting some examples of structural traffic patterns. It needs to be pointed out that while defining such patterns, care is taken to include edges between node classes in accordance with the description given in previous chapter. For example, edges between nodes of classes Pedestrian and Roadside are allowed but not between nodes of classes Pedestrian and Building. Even though the value and type of spatial relations between different node classes are not considered for defining traffic patterns, they motivate the inclusion of edges in traffic patterns.

Since attributes are not included to define traffic patterns, it allows us to model graph evolution in terms of change in graph structure (Topological Evolution) and consider the time domain $\mathbf{T}$ as discrete. Here, we consider the time-varying graph $\mathcal{G} = \{G_1, G_2, ..., G_n\}$, representing road traffic, in terms of discrete snapshots taken at *characteristic dates*, i. e. $G_i \neq G_{i+1}$, $1 \leq i \leq n$. This consideration is also in accordance with Topological Evolution described in the previous chapter.

In addition to graph $\mathcal{G}$, acting as *target graph*, we define *pattern graph* $\mathcal{H} = \{H_1, H_2, ..., H_m\}$, $m \leq n$, also as set of snapshots, representing a pattern to be detected in $\mathcal{G}$. Graph $\mathcal{H}$ acts as a template for the kind of pattern we are looking for in $\mathcal{G}$ in the sense that it outlines the temporal order which needs to be respected while performing subgraph isomorphism as well as provides the information about the classes into which the nodes are classified. However, it is possible that node labels in pattern and target graphs do not exactly match since pattern graph $\mathcal{H}$ is user-defined and target graph $\mathcal{G}$ is computed using real-world objects.

Having differentiated between the definitions of target and pattern graphs, let us now describe some pattern examples which can be

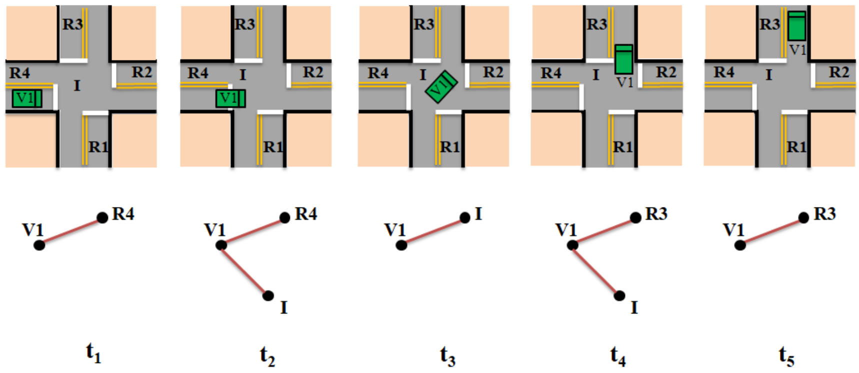


Figure 4.3: Vehicle crossing an intersection with corresponding graph structure varying at discrete timestamps

represented as $\mathcal{H}$ and given as input to the algorithm, described later, to detect subgraphs of $\mathcal{G}$ isomorphic to them.

4.3.1 *Pattern examples*

Examples given in this subsection represent some basic traffic situations and include both dynamic and static objects. At each ith timestamp t_i, graph snapshot H_i represents the traffic situation at t_i.

4.3.1.1 *Vehicle crossing an intersection*

One of the basic traffic situations is that of a vehicle crossing an intersection which is shown in Figure 4.3. In this situation, we consider five pattern graph snapshots which represent the location of the vehicle at a given instant. It is noteworthy that, at every time instant, graph topology changes as vehicle moves. When vehicle is on a road segment (at times t_1, t_2, t_4 and t_5), an edge between vehicle and corresponding road segment node exists in those graph snapshots. At t_3, vehicle is on the intersection, hence, there is an edge between vehicle and intersection nodes. At times t_2 and t_4, vehicle is shared between road segment and intersection.

4.3.1.2 *Pedestrian crossing the street*

Another traffic situation we consider is that of a pedestrian crossing a street while there is a vehicle approaching, as shown in Figure 4.4. At time t_1, the pedestrian is on the road side $FL1$ and at next time instant, on road $R1$ but not on the zebra crossing $M1$, which accounts for a dangerous traffic situation. We have shown the pattern graph snapshots for both time instants. They could be compared to the ones in Figure 4.5 where the pedestrian crosses the road on zebra crossing, in which case, at time t_2, there is an edge between node representing pedestrian and the one representing zebra crossing.

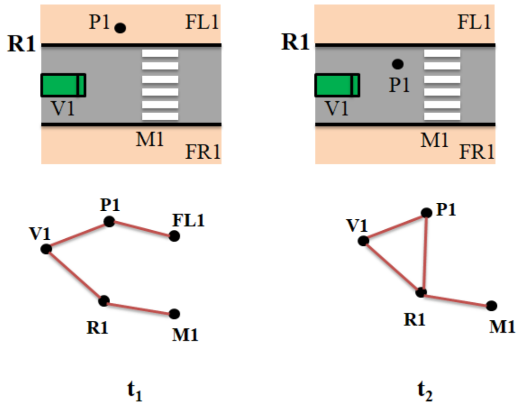


Figure 4.4: Pedestrian crossing the street but not from zebra crossing leading to dangerous traffic situation

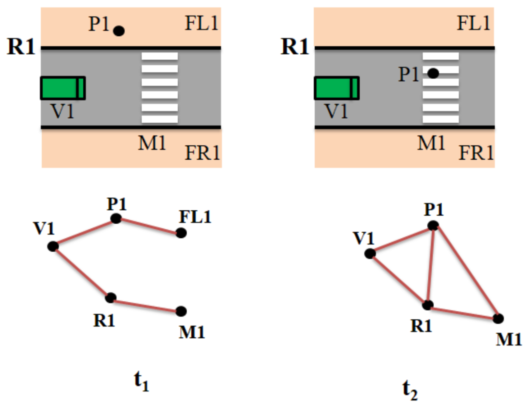


Figure 4.5: Pedestrian crossing the street from zebra crossing making this situation not dangerous

Thus, pattern graph topology helps in differentiating dangerous from non-dangerous traffic situations.

4.3.1.3 *Vehicle crossing intersection while pedestrian crossing the street*

This situation combines both cases described above. In this case, we consider two possibilities. First, the vehicle having priority over the pedestrian, as shown in Figure 4.6. Second, the pedestrian having priority over the vehicle, shown in Figure 4.7. In the first case, the pedestrian has to wait for the vehicle to pass and hence the edge between pedestrian node $P1$ and road side node $FR1$ exists for four time instants t_1 to t_4. At time t_5, pedestrian starts crossing the road. On the other hand, in second case, the vehicle has to wait for the pedestrian to cross the road. Once again, by looking at the change in graph topology, these two traffic situations can be differentiated.

A closer look at structural traffic pattern examples described above reveals an interesting characteristic which is crucial for performing subgraph isomorphism using the proposed algorithm. Since contin-

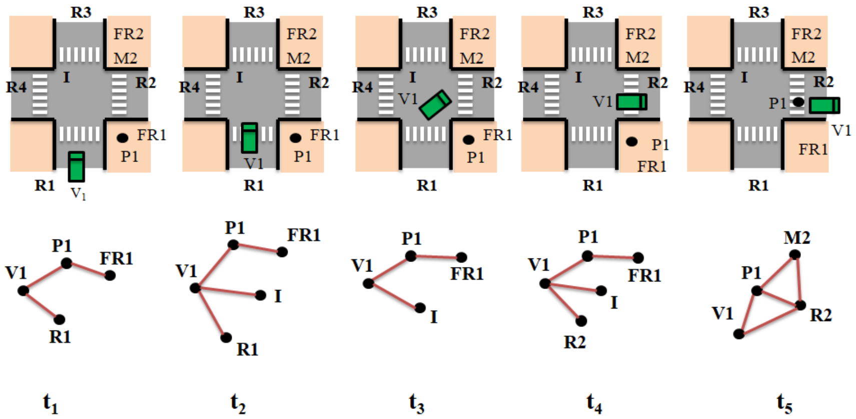


Figure 4.6: Vehicle crossing the intersection having priority while the pedestrian is waiting for the vehicle to pass

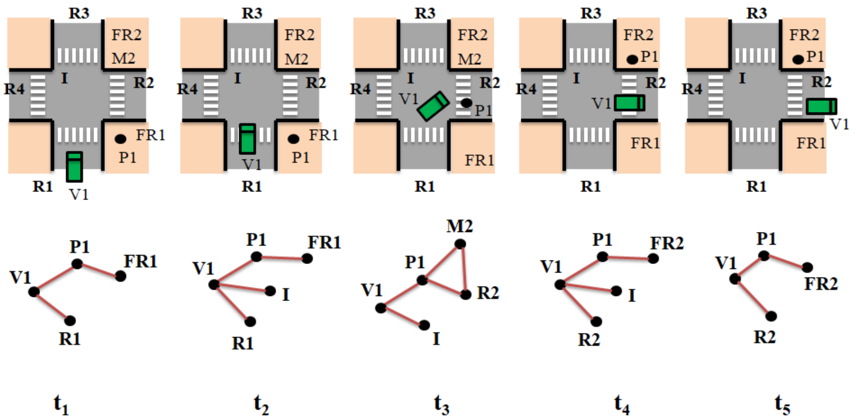


Figure 4.7: Vehicle crossing the intersection while the pedestrian has priority

uous movement of dynamic objects is discretized to form individual snapshots, there exists some nodes which are common in two adjacent pattern graph snapshots. For example, node $R4$, in Figure 4.3, is common in snapshots at times t_1 and t_2 and node I is common in snapshots at times t_3 and t_4 in Figure 4.7.

The algorithm maps pattern graph snapshots while respecting their temporal order. In addition, nodes common between snapshots H_i and H_{i+1}, $1 \leq i \leq m$ are mapped to some nodes of $G_j \in \mathcal{G}, 1 \leq j \leq n$ while mapping H_i to G_j, therefore, they need not be mapped again while mapping H_{i+1}. This property reduces the size of the state space while finding potential mappings.

Let us now discuss the proposed algorithm for subgraph isomorphism using dynamic pattern and target graphs in detail.

4.3.2 *Algorithm for pattern detection*

As mentioned before, the algorithm we propose is based on another algorithm called VF3 from Carletti et al., 2018, which is developed for subgraph isomorphism for static pattern and target graph pair. This algorithm uses a tree-based state-space representation, where each state represents a partial mapping between pattern and target graphs and if a state is found to be consistent, another state is added as a node in the tree representing the state-space. If a state is found to be inconsistent, then the algorithm backtracks to a previously consistent state and checks another potential node mapping. Starting from an empty state, the algorithm performs a depth-first search to add states and check their consistency. In the end, a consistent state which includes mappings between all nodes of pattern graph to some nodes of target graph, is considered to be the solution. This algorithm is designed to find all mappings between given pattern and target graph pairs.

We extend this algorithm to map dynamic pattern graph with a dynamic target graph. Pattern graph $\mathcal{H}$ defines the user-defined template which is to be searched in the road traffic represented using spatio-temporal graph $\mathcal{G}$, acting as target graph, described in the previous chapter. As mentioned before, both graphs are considered in terms of their static snapshots i. e. $\mathcal{G} = \{G_1, ..., G_n\}$ and $\mathcal{H} = \{H_1, ..., H_m\}$, where G_j represents a target graph snapshot at time instant j and H_i represents pattern graph snapshot at time instant i. Node set and edge set of snapshot G_j is written as $V(G_j)$ and $E(G_j)$ respectively. Similar notation is used for pattern graph snapshots and induced subgraphs. Nodes of both pattern and target graphs are classified into different classes (represented by the first character of their labels), and both pattern and target graphs contain nodes with unique node labels. While performing subgraph isomorphism, node class information is taken into account, i.e., nodes matched with each other belong to the same class.

We apply VF3 recursively to find all mappings of $\mathcal{H}$ in $\mathcal{G}$. However, we are interested in solutions which preserve node labels of mapped subgraphs of target graph over time. Since $\mathcal{H}$ is a template representing connections between certain object classes, finding it (subgraph) isomorphic to $\mathcal{G}$ means that connections between those object classes are found and node labels of matched nodes of target graph are preserved over time. Furthermore, we consider that there exist some snapshots of target graph to which no snapshot of pattern graph is mapped, i. e. they represent "noise". This makes the proposed algorithm suitable for real implementation with the possibility of choosing a temporal detail for discretizing time-varying road traffic graph into snapshots and then performing post-processing to detect a user-defined pattern.

Figure 4.8 shows an example where the pattern graph $\mathcal{H}$, representing vehicle crossing an intersection (Figure 4.3), is detected in the target graph $\mathcal{G}$. Number of snapshots of $\mathcal{H}$ is five and that of $\mathcal{G}$ is eight. Here, graphs are mapped as follows: $H_1 \rightarrow G_1$, $H_2 \rightarrow G_2$, $H_3 \rightarrow G_3$,

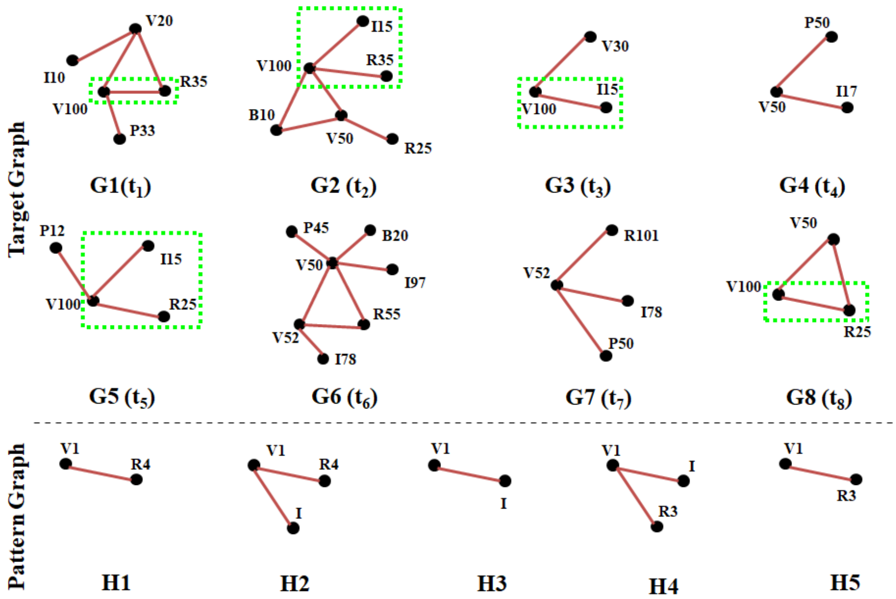


Figure 4.8: Example showing snapshots of target graph $\mathcal{G}$ and pattern graph $\mathcal{H}$ with detected pattern highlighted in $\mathcal{G}$. Pattern graph acts as template whose structure and node classes are taken into account when performing subgraph isomorphism.

$H_4 \to G_5$ and $H_5 \to G_8$, while nodes are mapped as: $V1 \to V100$, $R4 \to R35$, $I \to I15$ and $R3 \to R25$. Notice that while performing subgraph isomorphism, only node classes and structure of each H_i is taken into account. In addition, temporal order of $\mathcal{H}$ is maintained in the found solution and node labels of matched nodes of target graph are preserved over time. Some target graph snapshots (G_4, G_6 and G_7) represent "noise".

General strategy for our algorithm is as follows: Consider a pattern graph snapshot H_i for which we are trying to find an isomorphic subgraph of some target graph snapshot. If there are some nodes in H_i which were also present in H_{i-1} (considering continuous patterns, there would be some nodes common in adjacent pattern graph snapshots), it implies that these nodes were mapped while mapping H_{i-1} since VF3 maps all the nodes of a given pattern graph. For example, in Figure 4.8, nodes $V1$ and $R4$ of H_2 are already mapped while mapping H_1, and node $V1$ of H_3 is mapped while mapping H_2.

We suppose that H_{i-1} was mapped to some snapshot G_j of target graph, and we denote one such mapping as set $\mu_{(i-1)j} = \{(u, v) \mid u \in V(H_{i-1}),\ v \in V(G_j)\}$ (note that there could be multiple mappings between H_{i-1} and G_j). Suppose this mapping induced a subgraph $g_{(i-1)j}$ of G_j which was isomorphic to H_{i-1}. Since H_i occurs in future of H_{i-1}, it should be mapped to a target graph snapshot G_f in future of G_j, i.e. with $f > j$. To perform this mapping, we first look for the induced subgraph $g_{(i-1)j}$ in every G_f with $f > j$. Once we find a G_f which has $g_{(i-1)j}$ as its subgraph, we perform VF3 with H_i and G_f

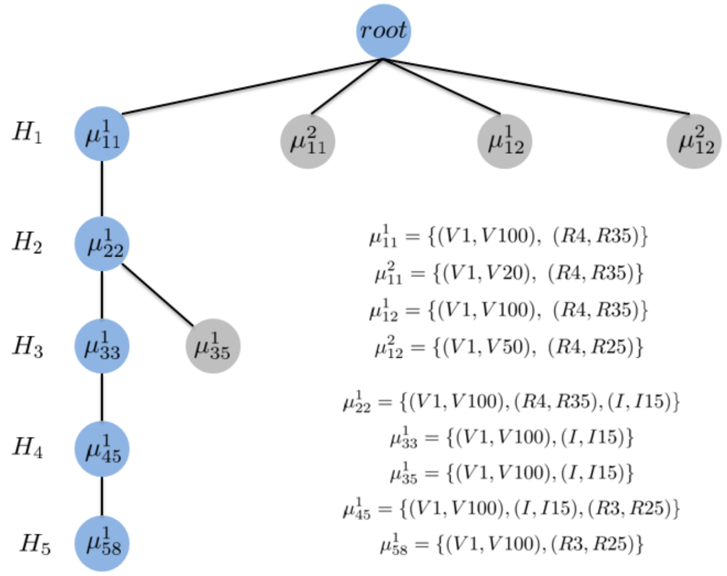


Figure 4.9: *Tree* maintained throughout the execution of the algorithm. Nodes coloured in blue represent the root-to-leaf path which gives the complete solution for subgraph isomorphism between $\mathcal{G}$ and $\mathcal{H}$ shown in Figure 4.8

as inputs. However, now, we start VF3 with a non-empty initial state which represents the previous mapping $\mu_{(i-1)j}$, thus reducing the size of the state space to be searched. To do this, we redefine the feasibility sets for both H_i and G_f as the set of nodes of H_{i-1} and G_j respectively, included in mapping $\mu_{(i-1)j}$. We recall that feasibility set is the set of nodes of a graph which are neighbors of nodes already included in the mapping, and is defined for both pattern and target graphs to implement VF3 (see Carletti et al., 2018). It is also possible that $H_i \subset H_{i-1}$, which means that all nodes of H_i were mapped while mapping H_{i-1} (for example $H_3 \subset H_2$ and $H_5 \subset H_4$ in Figure 4.8). In this case, there is no need to apply VF3 again but we still have to look for a suitable G_f with $f > j$ to maintain the temporal order of the final solution. The costliest operation in this algorithm is for mapping the first pattern graph snapshot since, in this case, VF3 from an empty initial state has to be applied between H_1 and G_j, $1 \leq j \leq n - m + 1$. Note that we do not need to iterate through any G_j, $j > n - m + 1$ since a mapping found in such target graph snapshots would not lead to a complete solution.

Given that VF3 finds all mappings between a pattern and a target graph, to keep track of these mappings we maintain a separate tree structure *Tree* whose nodes represent individual mappings. The tree structure for the example shown in Figure 4.8 is given in Figure 4.9. Mappings computed, whether by applying VF3 or not, by looking for induced subgraph $g_{(i-1)j} \subset G_j$ in future target graph snapshots G_f, are stored as child nodes of the node representing $\mu_{(i-1)j}$ in *Tree*. All mappings between a pattern graph snapshot H_i and any target graph snapshot G_j are stored at depth i in *Tree*, having empty *root* node at depth zero. The set of mappings lying on every root-to-leaf path of length $m + 1$, highlighted in blue in Figure 4.9, in *Tree* gives the final solution, where all pattern graph snapshots are mapped in a continuous manner. Functions implementing the proposed algorithm are given as Algorithms 1 - 5.

Algorithm 1: Main function to find dynamic pattern graph $\mathcal{H} = \{H_1, ..., H_m\}$ in dynamic target graph $\mathcal{G} = \{G_1, ..., G_n\}$

```
Procedure main(𝒢, ℋ)
    Tree ← ∅
    root ← null
    for i := 1 to m do
        Solution_i ← ∅
        𝔾(H_i) ← ∅
        if i = 1 then
            Tree = MapFirstPatternGraph(i, Solution_i, 𝔾(H_i), Tree, n, m)
        end
        if i > 1 then
            Tree = MapNextPatternGraphs(i, Solution_i, 𝔾(H_i),
              Solution_(i−1), 𝔾(H_{i−1}), Tree, n)
        end
    end
    P ← ∅                                  // set of root-to-leaf paths
    P = GetRootLeafPaths(Tree)
    forall rlp ∈ P do
        if len(rlp) = m + 1 then
            μ_ℋ𝒢 = {μ | ∀ μ ∈ rlp}
            /* μ_ℋ𝒢 is the complete solution */
        end
    end
end
```

Algorithm 1 is the main function which takes the pattern and target graph sequence as input and maintains a tree structure *Tree* to keep track of all the mappings for each H_i. It initialises *Tree* with a null *root* node. Once all pattern graph snapshots are mapped, *Tree* is iterated to find the entire instance of $\mathcal{H}$ as the final solution which is given as a root-to-leaf path of length $m + 1$ in *Tree*. The set $Solution_i$ stores all the found mappings between H_i and any target graph snapshot G_j and set $\mathbb{G}(H_i)$ stores all the induced subgraphs corresponding to each mapping. The main function calls `MapFirstPatternGraph` and `MapNextPatternGraphs` depending on index i of $\mathcal{H}$.

`MapFirstPatternGraph` function (Algorithm 2) applies VF3 recursively and stores the mapping between H_i and a particular G_j in set $Solution_{ij}$ and corresponding subgraphs in set $\mathbb{G}_{ij}$. The mapping μ_{ij} is given as set of pair of nodes $\{(u, v) \mid u \in V(H_i),\ v \in V(G_j)\}$. Note that since VF3 returns all found mappings, it is possible that H_i gets

Algorithm 2: Function to map H_1. It finds all snapshots of $\mathcal{G}$ having subgraphs isomorphic to H_1. This is done by applying VF3 recursively.

Function *MapFirstPatternGraph(i, $Solution_i$, $\mathbb{G}(H_i)$, Tree, n, m)*
  for j := 1 *to n - m + 1* **do**
    $Solution_{ij} \leftarrow \emptyset$
    $\mathbb{G}_{ij} \leftarrow \emptyset$
    if $VF3(H_i, G_j)$ = *true* **then**
      $Solution_{ij} \leftarrow \{\mu_{ij} \mid \mu_{ij} : V(H_i) \leftrightarrow V(g_{ij}),\ g_{ij} \subset G_j\}$
      `/*` $Solution_{ij}$ `stores all mappings between` H_i `and` G_j `*/`
      $Child(Tree.access(root)) \leftarrow \mu_{ij}$ `// make` μ_{ij} `child of` *root* `node`
      $Solution_i \leftarrow Solution_i \cup Solution_{ij}$
      $\mathbb{G}_{ij} \leftarrow \{g_{ij} \mid g_{ij} \subset G_j\}$
      `/*` $\mathbb{G}_{ij}$ `is the set of subgraphs of` G_j `isomorphic to` H_i `*/`
      `/*` $len(Solution_{ij}) = len(\mathbb{G}_{ij})$ `*/`
      $\mathbb{G}(H_i) \leftarrow \mathbb{G}(H_i) \cup \mathbb{G}_{ij}$
    end
  end
  return *Tree*
end

mapped multiple times to a given G_j, hence, induces multiple subgraphs of G_j. As a result, $len(Solution_{ij})$ and $len(\mathbb{G}_{ij})$ could be greater than 1. Furthermore, all found mappings for H_i are stored as child nodes of *root* in *Tree*.

`MapNextPatternGraphs` (Algorithm 3) function first checks if there are any common nodes between H_i and H_{i-1}, since such nodes are already mapped. These nodes are stored in the set Nodes Already Mapped (NAM). The remaining nodes are stored in set Nodes Not Mapped (NNM). If there are un-mapped nodes in H_i then *FindNewMapping* is called, otherwise *UsePreviousMapping* is called.

`FindNewMapping` (Algorithmn 4) calculates the feasibility set $\mathbb{F}(H_i)$ for H_i and chooses a previous mapping $\mu^a_{(i-1)j}$, with a being its index, between H_{i-1} and a G_j. This mapping is updated to include only the node pairs having the nodes present in $V(H_i)$ as first part and their corresponding nodes in $V(G_j)$ as second part, and is written as $\overline{\mu^a_{(i-1)j}}$. The subgraph induced by this mapping is given as $\overline{g^a_{(i-1)j}}$. This subgraph is searched in every $G_f,\ f > j$ and if found, VF3 is applied with new feasibility sets $\mathbb{F}(H_i)$ and $\mathbb{F}(G_f)$ (feasibility set calculated for G_f). VF3 requires calculating the node exploration sequence for defining an order to iterate through the nodes of pattern graph. Since in this function, VF3 is initialised from a non-empty state which contains the partial mapping between H_i and G_f, nodes included in the mapping at this initial state are not considered in node exploration

Algorithm 3: Function to map remaining snapshots of $\mathcal{H}$. Nodes common between H_i and H_{i-1} are already mapped. Remaining nodes are mapped by $FindNewMapping$ function

Function *MapNextPatternGraphs(i,* $Solution_i$, $\mathbb{G}(H_i)$, $Solution_{(i-1)}$, $\mathbb{G}(H_{i-1})$, *Tree, n)*
  $NAM_i \leftarrow V(H_i) \cap V(H_{i-1})$
  $NNM_i \leftarrow V(H_i) - NAM_i$
  if $NNM_i \neq \emptyset$ **then**
    Tree = FindNewMapping(i, NAM_i, NNM_i, $Solution_i$, $\mathbb{G}(H_i)$, $Solution_{(i-1)}$, $\mathbb{G}(H_{i-1})$, *Tree*, n)
  end
  else
    Tree = UsePreviousMapping(i, $Solution_i$, $\mathbb{G}(H_i)$, $Solution_{(i-1)}$, $\mathbb{G}(H_{i-1})$, *Tree*, n)
  end
  return *Tree*
end

sequence, which reduces the state space to be explored to map unmapped nodes. Similar to Algorithm 2, *Tree* is updated with suitable child nodes of the considered mapping $\mu^a_{(i-1)j}$.

`UsePreviousMapping` (Algorithm 5), on the other hand, is called when there are no new nodes in H_i to be mapped. It just looks for the previous mapping $\mu^a_{(i-1)j}$ in $G_f, f > j$ and updates *Tree*.

Spatio-temporal graph $\mathcal{G}$, described in the previous chapter, acts as target graph in which the user-defined structural traffic pattern, given as dynamic graph $\mathcal{H}$, is to be detected. Topological Evolution, also described in previous chapter, for both pattern and target graphs is considered, i. e. value and type of spatial relations is not considered to define pattern graph and neither is taken into account in case of target graph. Both pattern and target graphs are considered in terms of static snapshots, where snapshots of target graph are taken at *characteristic dates*.
Pattern graph $\mathcal{H}$ represents different types of traffic situations, and by formalizing change in graph structure, evolution of traffic is described. Some pattern graphs are also helpful in differentiating dangerous from non-dangerous traffic situations.
An algorithm, which extends existing algorithm VF3, to the case of dynamic graphs is proposed. It exploits the continuous nature, i. e. some nodes in adjacent pattern graph snapshots being common, of pattern graphs to reduce the size of the search space. In the proposed algorithm, pattern graph acts as a template describing temporal order of, and node classes considered in, detected subgraphs of target graph.

Algorithm 4: Function to map un-mapped nodes

Function *FindNewMapping(i,* NAM_i, NNM_i, $Solution_i$, $\mathbb{G}(H_i)$, $Solution_{(i-1)}$, $\mathbb{G}(H_{i-1})$, *Tree, n)*
    $\mathbb{F}(H_i) \leftarrow \{u \mid u \in NNM_i, \exists\, u' \in NAM_i,\ (u,\ u') \in E(H_i)\}$
    for $j := 1$ *to* n **do**
        for $a := 1$ *to* $len(\mathbb{G}_{(i-1)j})$ **do**
            if $g^a_{(i-1)j} \in \mathbb{G}_{(i-1)j}$ **then**
                $\mu^a_{(i-1)j} \leftarrow Solution^a_{(i-1)j}$
                `/* previous mappings are stored in` $Solution_{(i-1)j}$ `*/`
                $\overline{\mu^a_{(i-1)j}} \leftarrow \{(p,\ q) \in \mu^a_{(i-1)j} \mid p \in V(H_i),\ q \in V(g^a_{(i-1)j})\}$
                $\overline{g^a_{(i-1)j}} \leftarrow (q,\ (q,\ q')) \mid q,\ q' \in V(g^a_{(i-1)j})$
                for $f > j$ *to* n **do**
                    if $\overline{g^a_{(i-1)j}} \subset G_f$ **then**
                        $\mathbb{F}(G_f) \leftarrow \{p \mid p \in V(G_f),\ \exists\ p' \in V(g^a_{(i-1)j}),\ (p,\ p') \in E(G_f)\}$
                        if $VF3(H_i, G_f, \overline{\mu^a_{(i-1)j}}, \mathbb{F}(H_i), \mathbb{F}(G_f))$ = *true* **then**
                            $Solution_{if} \leftarrow \{\mu_{if} \mid \mu_{if} : V(H_i) \leftrightarrow V(g_{if}),\ g_{if} \subset G_f\}$
                            $Child(Tree.access(\mu^a_{(i-1)j})) \leftarrow \mu_{if}$
                            $Solution_i \leftarrow Solution_i \cup Solution_{if}$
                            $\mathbb{G}_{if} \leftarrow \{g_{if} \mid g_{if} \subset G_f\}$
                            $\mathbb{G}(H_i) \leftarrow \mathbb{G}(H_i) \cup \mathbb{G}_{if}$
                        end
                    end
                end
            end
        end
    end
    return *Tree*
end

4.4 ALGORITHM BENCHMARKING

Having described the algorithm for performing subgraph isomorphism for time-varying pattern and target graphs, let us now take a look at some benchmarking results we got while implementing the said algorithm. Since the spatio-temporal graph model of road traffic we proposed in this thesis is not yet implemented, to perform algorithm benchmarking, we developed a random graph generator to generate pairs of random dynamic pattern and target graphs. It allows us to vary graph parameters such as *size (N)*, *density (d)*, *number of node classes (c)* and *number of time stamps (t)* for both pattern and target graphs, enabling us to verify the performance of the algorithm for different values of these parameters. Thanks to random graph generator, we are able to test the algorithm with parameter values

Algorithm 5: Function to use previous mapping information in case there are no new nodes in H_i

Function *UsePreviousMapping(i, $Solution_i$, $\mathbb{G}(H_i)$, $Solution_{(i-1)}$, $\mathbb{G}(H_{i-1})$, Tree, n)*
- **for** $j := 1$ *to* n **do**
  - **for** $a := 1$ *to* $len(\mathbb{G}_{(i-1)j})$ **do**
    - **if** $g^a_{(i-1)j} \in \mathbb{G}_{(i-1)j}$ **then**
      - $\mu^a_{(i-1)j} \leftarrow Solution^a_{(i-1)j}$
      - $\overline{\mu^a_{(i-1)j}} \leftarrow \{(p, q) \in \mu^a_{(i-1)j} \mid p \in V(H_i), q \in V(g^a_{(i-1)j})\}$
      - $\overline{g^a_{(i-1)j}} \leftarrow (q, (q, q')) \mid q, q' \in V(g^a_{(i-1)j})$
      - **for** $f > j$ *to* n **do**
        - **if** $\overline{g^a_{(i-1)j}} \subset G_f$ **then**
          - $\mu^a_{if} \leftarrow \overline{\mu^a_{(i-1)j}}$
          - $Solution_{if} \leftarrow \{\mu^a_{if} \mid \mu^a_{if} : V(H_i) \leftrightarrow V(g^a_{if}), g^a_{if} = \overline{g^a_{(i-1)j}}\}$
          - $Child(Tree.access(\mu^a_{(i-1)j})) \leftarrow \mu^a_{if}$
          - $Solution_i \leftarrow Solution_i \cup Solution_{if}$
          - $\mathbb{G}_{if} \leftarrow \{g^a_{if}\}$
          - $\mathbb{G}(H_i) \leftarrow \mathbb{G}(H_i) \cup \mathbb{G}_{if}$
        - **end**
      - **end**
    - **end**
  - **end**
- **end**
- return *Tree*

end

which might not otherwise occur in implemented spatio-temporal graph or in considered pattern graph. For example, we perform tests with target graph size upto 700 nodes and pattern graph size upto 20 nodes, but in real case we might have smaller pattern and target graphs.

In this section, we describe the random graph generator we developed for generating random dynamic pattern and target graphs using uniform node labeling for both graphs. Then we will present the results of algorithm benchmarking. We plot average runtime and population standard deviation over 50 graph pairs while varying each parameter. Note that we consider total time taken by the algorithm, not just matching time, excluding the time required to read the graphs from files.

4.4.1 *Random graph generator*

Random graph generator is developed using the Java library called GraphStream, proposed in Dutot et al., 2007[154]. This library is specifi-

[154] Dutot et al. (2007). *GraphStream: A Tool for bridging the gap between Complex Systems and Dynamic Graphs*

Notation	Description	Value
N_t	Number of nodes in target graph	$\{100, 300, 700\}$
c_t	Number of node classes in target graph	$\{3, 6, 10\}$
d_t	Density of target graph snapshots	$\{0.1, 0.2, 0.3, 0.4\}$
t_t	Number of time stamps (snapshots) in target graph	$\{3, 5, 7, 10\}$
N_p	Number of nodes in pattern graph	$\{5, 10, 20\}$
c_p	Number of node classes in pattern graph	$\{3, 5, 8, 10\}$
d_p	Density of pattern graph snapshots	$\{0.2, 0.4, 0.6, 0.8\}$
t_p	Number of time stamps (snapshots) in pattern graph	$\{3, 5, 7, 10\}$

Table 4.1: Notations with their description and values used in our experiments

cally developed for working with dynamic graphs and supports an event-based graph model, where event corresponds to addition or removal of nodes and edges. In addition, events occurring at a time instant can be grouped together. Notations for graph parameters and their values used in our experiments are given in Table 4.1.

4.4.1.1 *Generating pattern graphs*

We start with generating the pattern graph. Since we rely on a snapshot-based model, we consider events occurring in each snapshot separately. First, the node set of size N_p is generated where each node is labeled randomly out of set of labels of size c_p. For generating random graphs, we consider the alphabets of English language as labels. At each time step and for a given value of d_p, edges are added randomly with uniform probability and nodes connected with edges are marked as *visited*. In addition, some edges are randomly removed for each time step $t > 1$. In order to ensure that pattern graph represents a continuously evolving connected pattern, the set of *visited* nodes at each time instant is used while adding or removing the edges. In the end, the largest connected component is taken as the pattern graph snapshot at every time instant since pattern graph needs to be a connected graph to apply the algorithm VF3.

4.4.1.2 *Generating target graphs*

The generation of target graph snapshots starts in the same way as pattern graph snapshots. First the node set of size N_t is generated where the nodes are labeled uniformly. However, in this case, the nodes of the pattern graph generated before are also included, and marked as *visited*, in the node set of target graph, since our objective is to ensure that at least one instance of the considered pattern exists in the target graph. Similar to pattern graph, at each time step of target graph, edges are added while keeping density d_t for each snapshot. However, in this case, the continuity of the graph is not taken into account and at each time step $t > 1$, all previous edges are removed before adding new edges. This ensures that pattern graph snapshots do not map to consecutive target graph snapshots. The idea here is to have some snapshots which represent *noise* in the target graph sequence, i. e. they will not be matched to any snapshot of the pattern graph sequence. In addition, node permutation applied to target graph nodes ensures that pattern and target graphs have different node ordering. In the end, all nodes and edges for every time stamp are taken into account, as a result, we can have target graph snapshots which are disconnected graphs.

4.4.2 *Experiments*

We performed our experiments on a single node in the node cluster called Myria maintained by CRIANN (more information about Myria can be found at http://www-tech.criann.fr/calcul/tech/myria-doc/guide-util/). It employs SLURM resource manager to allocate nodes at run time. Although a Myria node has large number of physical cores, for our experiments, we demanded three cores. To execute a serial algorithm in parallel, we used the library OpenMP and its loop worksharing construct using which a `for` loop can be executed in parallel. We parallelize `for` loops in line 2 of Algorithm 2, in line 3 of Algorithm 4 and line 2 of Algorithm 5. Since we asked for three cores, we created three OpenMP threads. However, sometimes during our experiments, three physical cores were not available on the allocated node, leading to created threads being waiting their turn to execute, hence, some discrepancies in our experiments.

The code for pattern detection was written in C++ and executed on Linux operating system using Intel compiler (version 17.0.1.132) with level-3 optimisation. As mentioned before, we varied graph parameters and computed average total runtime and standard deviation for 50 graph pairs for each of our experiments. It needs to be pointed out that we plotted elapsed wall time computed using `omp_get_wtime()` OpenMP routine.

4.4.2.1 *Changing size and density of target graphs*

For this experiment, we varied the size of target graph as $N_t = 100$, $N_t = 300$ and $N_t = 700$ and density as $d_t = 0.2$, $d_t = 0.3$ and $d_t = 0.4$, for fixed number of node classes of target graph $c_t = 10$ and snapshots $t_t = 3$. We fixed pattern graph parameters as: $N_p = 10$, $c_p = 3$, $d_p = 0.6$, $t_p = 3$ and $N_p = 20$, $c_p = 3$, $d_p = 0.4$, $t_p = 3$ and plotted average time and standard deviation in Figure 4.10. Clearly, for larger and denser target graphs, the algorithm takes longer to complete, the behaviour coherent with VF3. Hence, target graph of appropriate size needs to be chosen for faster runtime in the real-world implementation.

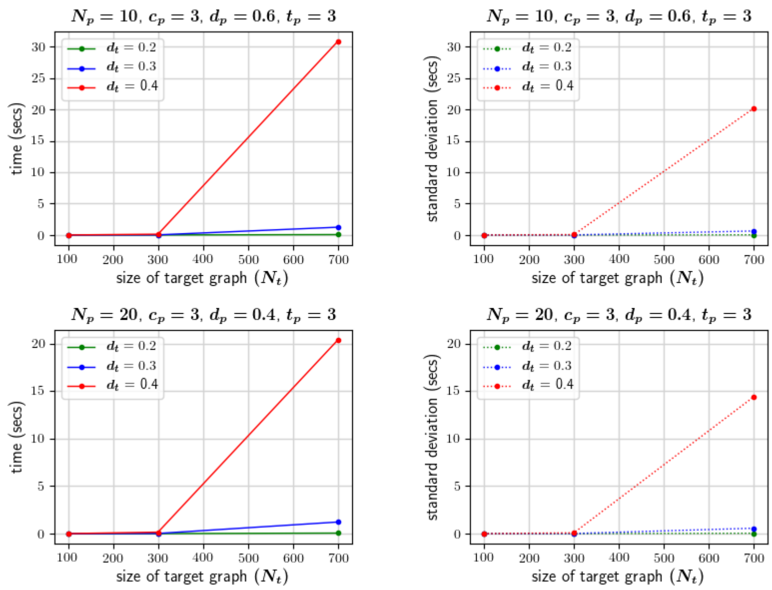


Figure 4.10: Changing size and density of target graphs

4.4.2.2 *Changing number of snapshots of target graphs*

Here, number of snapshots of target graph are varied as $t_t = 3$, $t_t = 5$, $t_t = 7$ and $t_t = 10$ with fixed number of node classes $c_t = 10$ and density $d_t = 0.1$. Pattern graph parameters are same as previous experiment. Average runtime increases with size and number of target graph snapshots, as shown in Figure 4.11. Here, two things need to be pointed out. First, for smaller target graph of size $N_t = 100$, increase in average runtime with increasing number of snapshots is not explicit, as seen in bottom-left plot. We believe this is due to the fact that experiments over 50 graph pairs are executed fairly quickly for parameter values $N_t = 100$ and $t_t = 10$ leading to an outlier. However, for larger target graph size, we can see the trend of increasing average runtime. We performed these experiments with target graph of size $N_t = 1500$ and the trend was clearer. But we haven't plotted the results to be coherent with other experiments where we vary the target graph size until $N_t = 700$. Plots for standard deviation show that dataset for a given set of parameters could vary from its mean due to the random nature of tested graphs.

Since with more number of target graph snapshots it takes more time for the algorithm to finish, for real-world implementation, this forces the user to chose appropriate temporal detail so that the desired pattern can be detected with comparatively lesser number of target graph snapshots.

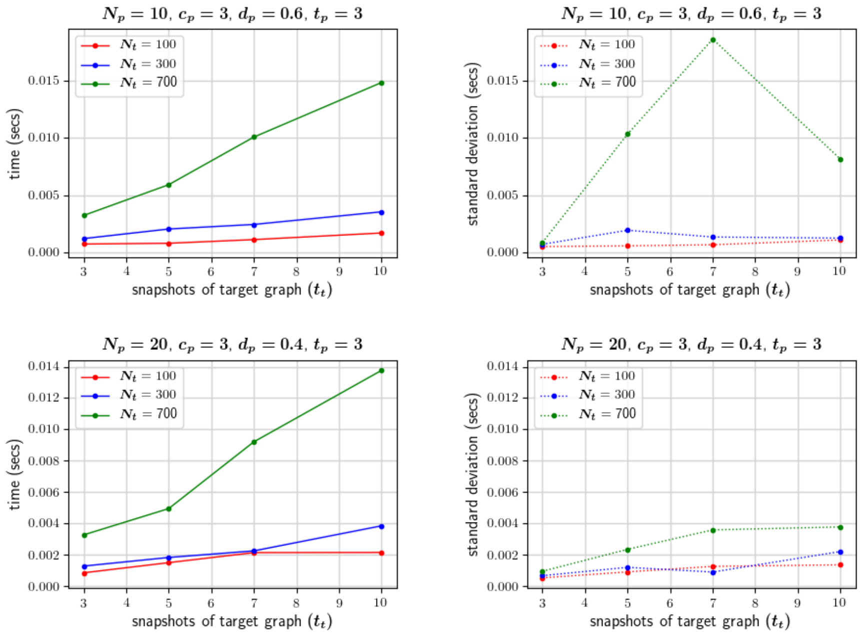


Figure 4.11: Changing number of snapshots of target graphs

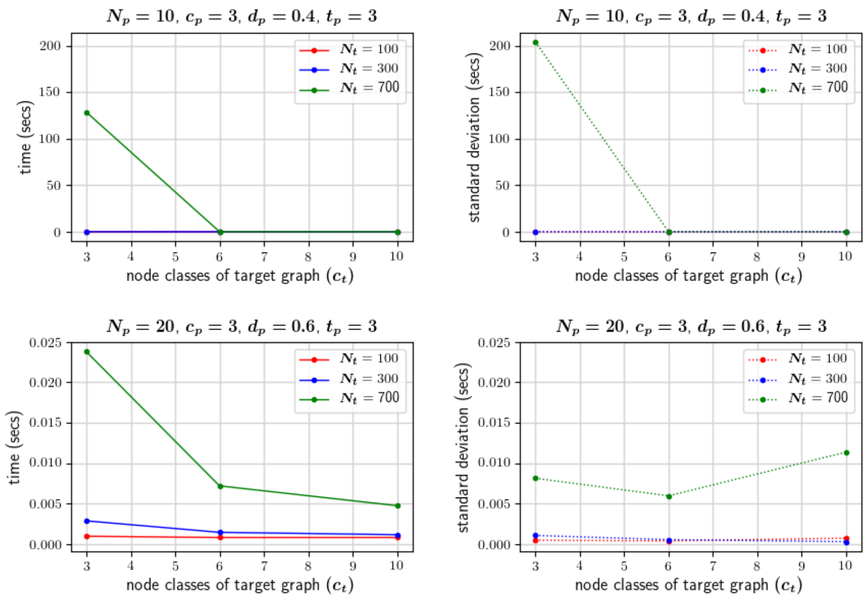


Figure 4.12: Changing number of node classes of target graphs

4.4.2.3 *Changing number of node classes of target graphs*

VF3 takes into account the class information of nodes while looking for candidate node pairs to be verified for feasibility. It turns out that the number of classes into which the target graph nodes are classified plays an important role in computing average runtime. As seen from Figure 4.12, for a given target graph size, if more number of classes are considered, the average time taken by the algorithm reduces, since there exist less number of potential matches. For this experiment, we vary the number of target graph node classes as $c_t = 3$, $c_t = 6$ and $c_t = 10$ with fixed density $d_t = 0.1$ and number of snapshots $t_t = 3$. Parameters for pattern graph are: $N_p = 10$, $c_p = 3$, $d_p = 0.4$, $t_p = 3$ and $N_p = 20$, $c_p = 3$, $d_p = 0.6$, $t_p = 3$.

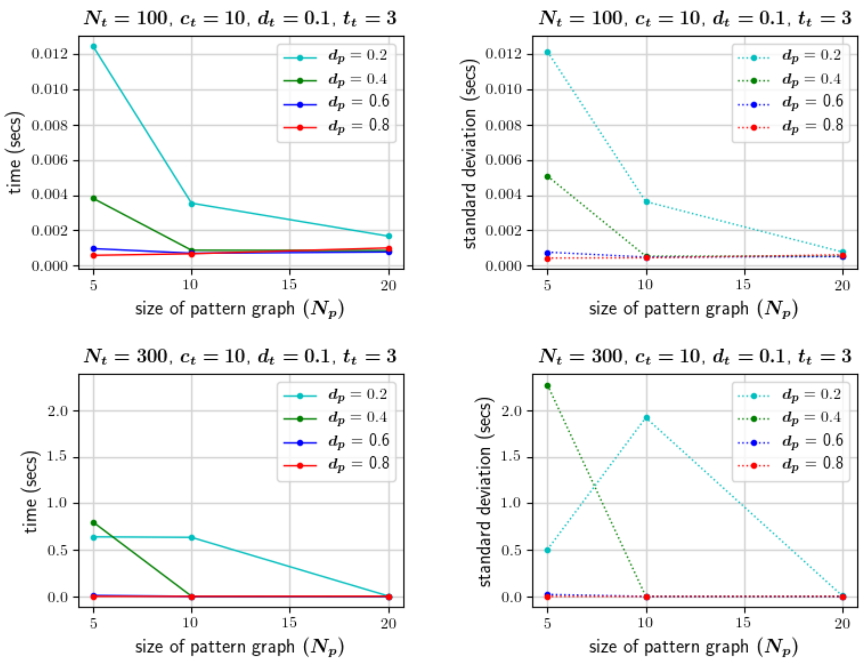


Figure 4.13: Changing size and density of pattern graphs

4.4.2.4 *Changing size and density of pattern graphs*

In our experiments, we also varied the parameters of pattern graphs to see their effect on algorithm runtime. We start by varying their size and density for fixed set of parameters of target graphs: $N_t = 100$, $c_t = 10$, $d_t = 0.1$, $t_t = 3$ and $N_t = 300$, $c_t = 10$, $d_t = 0.1$, $t_t = 3$. The results are plotted in Figure 4.13. It can be clearly seen from the figure that smaller and sparser patterns lead to higher average runtime for the algorithm, while time reduces for large and denser patterns. An explanation for this is that smaller and sparser patterns are more common and as their size and density increases, they become more particular, leading to less number of suitable matches in the target

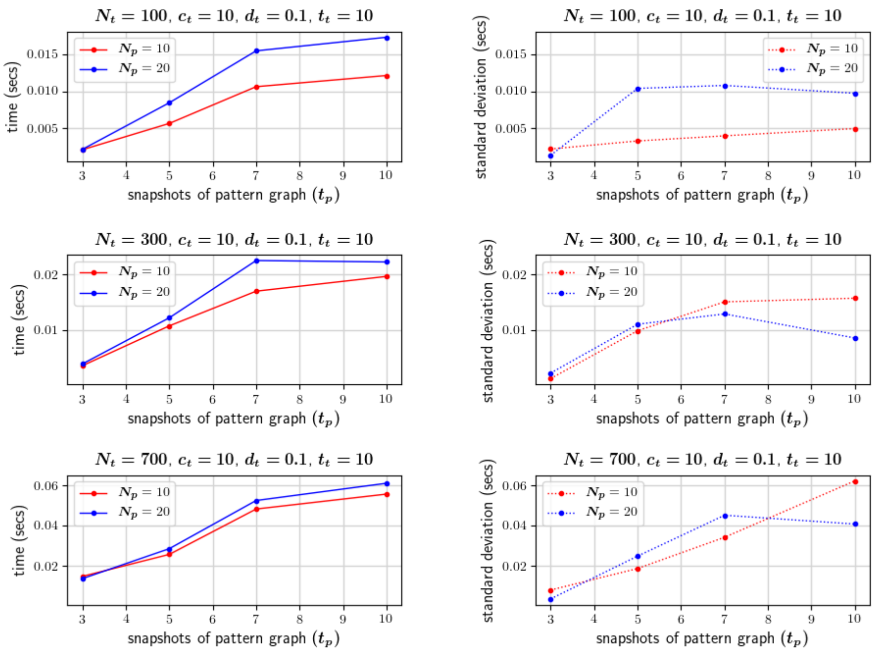


Figure 4.14: Changing number of snapshots of pattern graphs

graphs. We have plotted the results of these experiments by varying the size of pattern graphs as $N_p = 5$, $N_p = 10$ and $N_p = 20$, and their density as $d_p = 0.2$, $d_p = 0.4$, $d_p = 0.6$ and $d_p = 0.8$. However, we have only used target graphs of size $N_t = 100$ and $N_t = 300$ since, for $N_t = 700$, the experiments for 50 graph pairs with different parameters of pattern graph took too long to finish. In top-left plot of Figure 4.13, average time for $N_p = 20$ and $d_p = 0.8$ is a bit higher since there's an outlier which took more than three times the standard deviation to complete, leading to higher average value. Similar behaviour is observed in bottom-left plot for $N_p = 10$ and $d_p = 0.2$.

4.4.2.5 *Changing number of snapshots of pattern graphs*

Even though pattern graphs are user-defined and can have as many snapshots as required to model the pattern, higher number of snapshots of pattern graph leads to higher average runtime for the algorithm since more number of iterations are required to be performed. The results for this experiment are plotted in Figure 4.14, where we vary the number of pattern graph snapshots as $t_p = 3$, $t_p = 5$, $t_p = 7$ and $t_p = 10$ with density $d_p = 0.6$ and classes $c_p = 3$ for $N_p = 10$ and density $d_p = 0.4$ and classes $c_p = 3$ for $N_p = 20$. A peculiar behaviour is seen in the middle-left plot in Figure 4.14 for $t_p = 7$ and $N_p = 20$ due to an outlier.

4.4.2.6 *Changing number of node classes of pattern graphs*

Results for varying the number of node classes for pattern graph are shown in Figure 4.15. In general, they do not have a lot of effect on average time, as seen from plots for $N_p = 10$ and $N_p = 20$ for $N_t = 100$ and $N_p = 5$ and $N_p = 20$ for $N_t = 300$. However, it is a bit hard to judge their effect for uniform labeled pattern graphs of such a small size. We restrict ourselves to these values for size since, for real implementation, these values make sense as, at a given time, we would not want to take into account a lot of objects to represent the pattern graph snapshot.

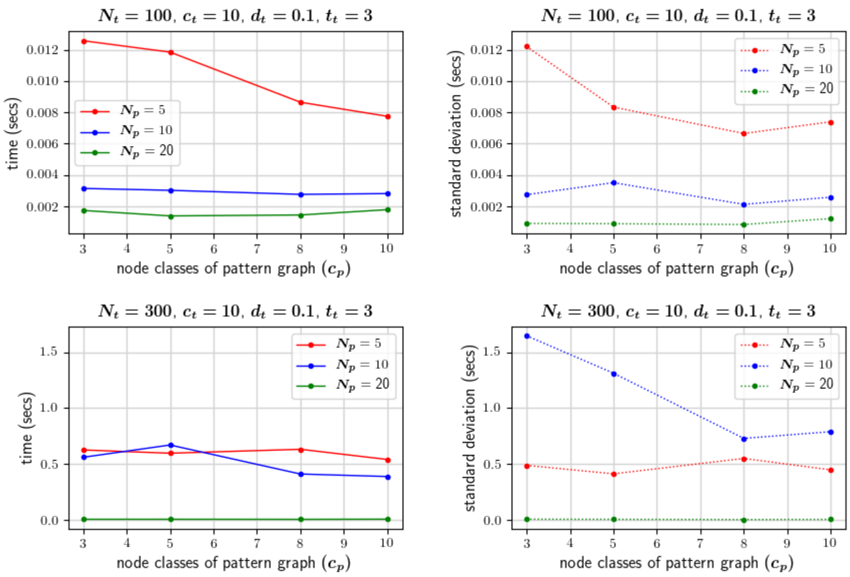


Figure 4.15: Changing number of node classes of pattern graphs

This chapter describes an important application of modeling road traffic using a spatio-temporal graph. Such a graph model describes the underlying structure of traffic by describing relations between various traffic constituents, something which is missing in traditional mathematical equations-based traffic models. By describing the fundamental structure of traffic using graph, we are able to extend the field of Traffic Pattern Detection, mainly focused on *statistical traffic patterns*, to also include *structural traffic patterns*. Spatio-temporal graph model of traffic allows us to detect such structural traffic patterns by applying the notion of *subgraph isomorphism* between itself, acting as target graph, and a time-varying pattern graph, representing a pattern template to be searched.
In this chapter, we have described various examples of structural patterns which could be detected in spatio-temporal graph for traffic, and proposed an algorithm which applies subgraph

isomorphism to detect all of such patterns occurring within it. In addition, we tested the algorithm with different set of parameters for both pattern and target graphs. This was made possible by developing a random graph generator which was able to generate random pattern and target graph pairs with different parameter values. We presented benchmarking results for the proposed algorithm which helped us in understanding the effect of these parameters on average runtime of the algorithm. It is seen that for larger and denser target graphs, runtime is higher, and for larger and denser pattern graphs, it gets lower. Furthermore, runtime increases with the number of snapshots of target and pattern graphs, and it reduces when more node classes are considered for target graphs. Number of node classes of pattern graphs have no clear effect on the runtime.

The obtained results of algorithm performance will guide the implementation of spatio-temporal graph representing road traffic since we will know the effect that graph parameters will have when the pattern detection algorithm will be applied. Hence, we would be able to skew the parameters of implemented graph to have lower runtime for the proposed algorithm.

5 Conclusion and Perspectives

Road traffic plays a significant role in our daily lives, hence, it has been the focus of research for past several decades, proven by the development of various mathematical models for its formalization and analyses. This thesis aims to carry forward the torch by bringing together research domains of Traffic Modeling and Spatio-Temporal Modeling on the basis of the notion that road traffic is inherently spatio-temporal. Hence, concepts proposed for describing spatio-temporal models could be applied in case of road traffic.

The main contributions of this thesis can be described as follows:

- First, we proposed a traffic model which takes a relative point of view on traffic as opposed to absolute point of view taken by existing traffic models. The relative characteristic of the proposed model becomes explicit by the definition of spatial relations between various traffic constituents included in it. Whereas existing traffic models mainly focus on quantitative traffic data, the proposed model focuses on qualitative knowledge represented in terms of these relations. The objective to consider qualitative knowledge is to be able to perform reasoning about interactions between traffic constituents and describe the behavior of dynamic objects under different traffic situations.

  Similar to existing traffic models which represent traffic at different scales (microscopic, macroscopic and mesoscopic), our model represents traffic from large scale of a city, having various road segments and intersections, as well as from small scale of a single road segment and/or intersection. Such change in the representation scale reduces the number of objects included in the model. Furthermore, we have considered that a bi-directional road segment can be divided in terms of its carriageways, a point of view useful for comparing traffic density and flow on each carriageway. In addition, a road segment can be segregated in terms of disjoint sectors which further reduces the number of included objects. Since road traffic evolves with time, temporal dimension needs to incorporated into the model. We consider time to be linear, totally ordered and continuous which can be discretized according to the considered temporal detail. The considered time domain consists of both time instants and intervals, where intervals are bounded by zero-duration instants.

The significant feature of our model is its formalization using a graph. Graphs are useful to model the structural aspect of a phenomenon or an object. In our case, since we are using the graph to model road traffic, we are able to represent the underlying structure of traffic, where various traffic constituents act as graph nodes and spatial relations are embedded using graph edges. To include the temporal dimension, we consider the graph to be time-varying and define two types of graph evolutions - Attribute Evolution, which concerns with variations in node and edge attributes, and Topological Evolution, which is related to variations in graph structure i. e. addition or removal of nodes and/or edges. Depending on the type of evolution, the time domain is considered to be continuous or discrete.

- Second, we redefine the meaning of traffic patterns by using the notion of underlying structure of traffic and considering structural patterns which take into account traffic constituents and their interactions. These patterns are represented using time-varying graphs and describe day-to-day traffic situations in terms of change in the structure of pattern graph (Topological Evolution) with time. Such patterns are also applicable in differentiating between dangerous and non-dangerous traffic situations, just by comparing the corresponding graph structures.

- Lastly, we proposed a pattern detection algorithm with the aim of detecting such structural traffic patterns in the spatio-temporal graph representing road traffic. This problem is formalized as the problem of *subgraph isomorphism* between time-varying pattern and target graphs. The proposed algorithm is tested using a random graph generator, which generates time-varying random pattern and target graph pairs, and varying the values of four graph parameters - size, density, number of node classes and number of snapshots - for both pattern and target graphs. Benchmarking results obtained in our experiments demonstrate the effect of these parameters on the average runtime of the algorithm and will be useful when the proposed spatio-temporal graph representing road traffic will be implemented.

This thesis is just a first step towards bringing together the fields of Traffic Modeling and Spatio-Temporal Modeling and applying Graph Theory to model traffic, there is still a long way to go. Some future research directions for our work are discussed below. We have divided them according to the time when we feel they should be carried out.

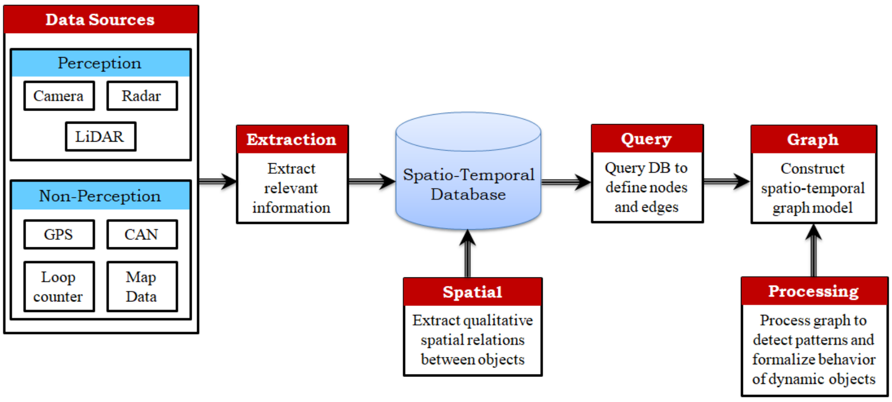


Figure 5.1: Global workflow for generating spatio-temporal graph to represent traffic

PERSPECTIVES FOR SHORT-TERM RESEARCH

Model implementation using simulated data

It would have been noticed by the reader that the proposed graph model is not implemented. Hence, implementation of the model is the next step. In fact, we made a choice between implementing the model and defining an application for using graphs to model road traffic and we chose to answer the question "what does graph bring to the table that is not included in existing traffic models?" and decided to exploit the structure of the graph to define traffic patterns and worked on the algorithm for pattern detection.

Even though the model itself is not implemented, the global workflow, shown in Figure 5.1, also described in the introduction of this thesis, represents different steps for implementation. As the first step, we have started developing a traffic simulator using an open-source modeling and simulation platform called GAMA (https://gama-platform.github.io) within which various traffic constituents and their movement patterns can be defined. This platform is developed for agent-based modeling applications but it has the ability to simulate real-world traffic behaviour and interactions between different agents. The motivation to start with simulated environment is to accelerate the process of constructing the graph without having to deal with **Data Sources** and **Extraction** modules of the workflow. Since it is a simulated environment, any number of data attributes for different agents can be considered which would act as node attributes for corresponding nodes in the graph. In addition, quantitative data about simulated environment can also be easily stored in the spatio-temporal database. Then using database extensions like PostGIS, spatial relations between simulated agents can be computed and the graph can be constructed using Java library GraphStream (http://graphstream-project.org/) as is done for developing random graph generator in Chapter 3.

Model implementation using real-world data

Once the graph is constructed using simulated data, we aim to replace GAMA platform with real-world data sources deployed in the city of Rouen, France. For this, the **Extraction** module from the workflow needs to clearly defined so that meaningful information about different objects detected using existing algorithms is stored in the database. The rest of the process for constructing the graph remains unchanged.

PERSPECTIVES FOR INTERMEDIATE-TERM RESEARCH

Classes of time-varying graphs

Once the proposed model is implemented using simulated or real data, one could focus on adding additional functionalities to the model. The time-varying graph model described in Casteigts et al., 2012[155] has been classified into various classes depending on the properties of the graph. These classes, defined for distributed problems, highlight the properties of the graph which could be applied to our model. For example, the class *Recurrence of edges*, which implies that if an edge appears once it can appear infinitely often, can be applied to traffic model and combined with edge-centric evolution to model appearance and disappearance of edges. Similarly, there are other classes of time-varying graphs described in Casteigts et al., 2012 and it should be interesting to see if they can provide new information about traffic.

[155] *Casteigts et al. (2012). Time-varying graphs and dynamic networks*

Point-of-views of graph evolution

We have talked about different point of views - node-centric, edge-centric and graph-centric - of graph evolution in Chapter 2. Although we have only focused on graph-centric evolution in terms of variations in graph structure and attributes of nodes and edges, focusing on node-centric evolution should also be interesting. For example, in case the node represents a vehicle, node-centric evolution would describe change in its neighbourhood over time using which its trajectory can be deduced. This idea is also described in Casteigts et al., 2012.

Temporal subgraph

Another concept which is implicit in the proposed model but not discussed is that of *temporal subgraph* where the lifetime of time-varying graph is reduced and only the nodes and edges present within it are considered. This is useful if we do not want to focus on the entire graph for the considered lifetime but only on the one corresponding to a sub-interval of the lifetime. Even for pattern detection, considering a temporal subgraph of the time-varying graph reduces the number of snapshots which have to searched.

Incorporating value of spatial relations in traffic patterns

Speaking of traffic patterns, we have not considered the value and type of spatial relations in their definition. Hence, the next step would be to include the value of different types of spatial relations and their variation over time to enrich patterns with meaningful semantics. Doing so would also help in detecting dangerous traffic situations and modeling the effect of spatial relations on the behaviour of dynamic objects.

Using underlying graph and footprint in traffic patterns

In addition, the concepts of *underlying graph* and *footprint* have not been combined with the problem of pattern detection. It would be interesting to see if using these concepts the algorithm for pattern detection could be improved.

Time window to detect meaningful patterns

The described algorithm for pattern detection is developed without having any constraints on time for detecting the pattern in target graph. It iterates through all target graph snapshots regardless of the number of snapshots. For now, we rely on the wisdom of the user for describing the number of snapshots for both target and pattern graphs. However, the algorithm can be modified to look for patterns within a given time window to consider them to be meaningful. For example, if one pattern snapshot is matched to a target graph snapshot at time t_5 and the next is matched to the snapshot at t_{50}, considering there are target graph snapshots for each time instant $t_5 < t < t_{50}$ and depending on the level of temporal detail considered, it is possible that the detected pattern is spread over the considered time domain and, hence, loses its importance. In such cases, the time window can restrict the number of target graph snapshots to make the detected pattern pertinent. This is also useful in case if dangerous traffic patterns are being detected.

PERSPECTIVES FOR LONG-TERM RESEARCH

Predicting pattern existence

The pattern detection algorithm proposed in this thesis requires that the spatio-temporal graph for road traffic, which acts as target graph, is already formalized. Hence, the stage of pattern detection in the workflow described in the introduction is a post-processing step applied on the graph to detect a given pattern. An extension of this idea is to be able to predict the existence of a pattern in the future snapshots of the target graph, provided that an instance of that pattern has been already detected in the target graph. Here, the idea is to compute some structural and attribute characteristics of the target graph which lead to the existence of that particular pattern. If such characteristics could be correctly defined and searched for in future target graph snapshots, it might give an idea if an instance of the given pattern would exist in the future.

Online pattern detection

Another direction of research could be to apply the pattern detection algorithm in real-time while the spatio-temporal graph for traffic is being formalized. This is an idea described for streaming graphs in Fan et al., 2011[156] where patterns are detected in an online fashion. It would be interesting to see if and how the processes of graph formalization and pattern detection can be combined.

[156] *Fan et al. (2011). Incremental Graph Pattern Matching*

Detecting time-varying structural patterns in other fields

Finally, since the need for pattern detection appears in many research fields, the proposed pattern detection algorithm can be applied in such fields just by modifying the definition of pattern and target graphs. One such example is the field of medical data analysis where patients' medical records, described using spatio-temporal graphs, can be analysed to detect disease patterns, also represented using time-varying graphs, by applying the proposed algorithm. This idea has been submitted as a Franco-Japanese joint project titled PHC SAKURA 2020.

Bibliography

Aldington, H. E., E. G. Beaumont, and J. S. Nicholl (1945)
"The Post-War Development of Road Motor Transport." In: *Proceedings of the Institution of Automobile Engineers* 39.4, pp. 361–384 (cited on page 9).

Allen, James F. (1983)
"Maintaining Knowledge About Temporal Intervals." In: *Communications of ACM* 26.11, pp. 832–843. ISSN: 0001-0782 (cited on page 25).

– (1984)
"Towards a general theory of action and time." In: *Artificial Intelligence* 23.2, pp. 123 –154. ISSN: 0004-3702 (cited on page 26).

Allen, James F. and Patrick J. Hayes (1985)
"A Common-sense Theory of Time." In: *Proceedings of the 9th International Joint Conference on Artificial Intelligence*. IJCAI'85. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., pp. 528–531. ISBN: 0-934613-02-8 (cited on pages 26, 65).

Armstrong, Marc P (1988)
"Temporality in spatial databases." In: *Proceedings of GIS/LIS'88: Accessing the World*. Vol. 2. American Society for Photogrammetry and Remote Sensing, pp. 880–889 (cited on pages 27, 28).

Basu, Prithwish et al. (2010)
"Modeling and analysis of time-varying graphs." In: *arXiv preprint arXiv:1012.0260* (cited on pages 62–64).

Batty, Michael (2003)
"Network Geography: Relations, Interactions, Scaling and Spatial Processes in GIS." In: *Centre for Advanced Spatial Analysis Working Paper Series*, 63:1–63:23. ISSN: 1467-1298 (cited on page 36).

Bhadra, Sandeep and Afonso Ferreira (2002)
Computing multicast trees in dynamic networks using evolving graphs. Tech. rep. RR-4531. INRIA (cited on page 61).

Biham, Ofer, A. Alan Middleton, and Dov Levine (1992)
"Self-organization and a dynamical transition in traffic flow models." In: *Physical Review A* 46 (10), R6124–R6127 (cited on page 12).

Bourrel, Emmanuel and Vincent Henn (2002)
"Mixing micro and macro representations of traffic flow: a first theoretical step." In: *Proceedings of the 9th meeting of the Euro Working*

Group on Transportation. Polytechnic University of Bari, Italy, pp. 610–616 (cited on page 15).

Brackstone, Mark and Mike McDonald (1999)
"Car-following: a historical review." In: *Transportation Research Part F: Traffic Psychology and Behaviour* 2.4, pp. 181 –196. ISSN: 1369-8478 (cited on page 12).

Bunke, Horst and Kaspar Riesen (2012)
"Towards the unification of structural and statistical pattern recognition." In: *Pattern Recognition Letters* 33.7, pp. 811 –825. ISSN: 0167-8655 (cited on page 77).

Carletti, Vincenzo et al. (2018)
"Challenging the Time Complexity of Exact Subgraph Isomorphism for Huge and Dense Graphs with VF3." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.4, pp. 804–818. ISSN: 0162-8828 (cited on pages 79, 81, 82, 85, 88).

Carletti, Vincenzo et al. (2018)
"Comparing performance of graph matching algorithms on huge graphs." In: *Pattern Recognition Letters*. ISSN: 0167-8655 (cited on page 81).

Casteigts, Arnaud (2018)
"Finding Structure in Dynamic Networks." In: *CoRR* abs/1807.07801 (cited on page 72).

Casteigts, Arnaud et al. (2012)
"Time-varying graphs and dynamic networks." In: *International Journal of Parallel, Emergent and Distributed Systems* 27.5, pp. 387–408 (cited on pages 64–67, 70–73, 106).

Castillo, J. M. del (1996)
"A car following model-based on the Lighthill-Whitham theory." In: *Proceedings of the 13th International Symposium on Transportation and Traffic Theory (ISTTT)*. Pergamon, N.Y., pp. 517–538 (cited on page 14).

Chandler, Robert E., Robert Herman, and Elliott W. Montroll (1958)
"Traffic Dynamics: Studies in Car Following." In: *Operations Research* 6.2, pp. 165–184 (cited on pages 11, 14, 35).

Chen, Juan et al. (2013)
"A survey of qualitative spatial representations." In: *Knowledge Engineering Review* 30.1, pp. 106–136 (cited on pages 20–23).

Chen, Peter Pin-Shan (1976)
"The Entity-relationship Model-Toward a Unified View of Data." In: *ACM Transactions on Database Systems (TODS)* 1.1, pp. 9–36. ISSN: 0362-5915 (cited on page 16).

Chen, W., F. Guo, and F. Wang (2015)
"A Survey of Traffic Data Visualization." In: *IEEE Transactions on Intelligent Transportation Systems* 16.6, pp. 2970–2984. ISSN: 1524-9050 (cited on page 27).

Cheng, Tao, James Haworth, and Jiaqiu Wang (2012)
"Spatio-temporal autocorrelation of road network data." In: *Journal of Geographical Systems* 14.4, pp. 389–413. ISSN: 1435-5949 (cited on page 51).

Chrisman, Nick (1977)
"Concepts of space as a guide to cartographic data structures." In: *First International Advanced Study Symposium on Topological Data Structures for Geographic Information Systems* (cited on page 19).

Claramunt, Christophe and Marius Thériault (1995)
"Managing Time in GIS: An Event-Oriented Approach." In: *Recent Advances in Temporal Databases*. Ed. by James Clifford and Alexander Tuzhilin. London: Springer London, pp. 23–42. ISBN: 978-1-4471-3033-8 (cited on page 32).

Claramunt, Christophe and Stephan Winter (2007)
"Structural Salience of Elements of the City." In: *Environment and Planning B: Planning and Design* 34.6, pp. 1030–1050 (cited on pages 37, 40).

Clementini, Eliseo (2013)
"Directional relations and frames of reference." In: *GeoInformatica* 17.2, pp. 235–255. ISSN: 1573-7624 (cited on pages 22, 51).

Clementini, Eliseo, Paolino Di Felice, and Daniel Hernández (1997)
"Qualitative representation of positional information." In: *Artificial Intelligence* 95.2, pp. 317 –356. ISSN: 0004-3702 (cited on page 22).

Clifford, James and David S. Warren (1983)
"Formal Semantics for Time in Databases." In: *ACM Transactions on Database Systems* 8.2, pp. 214–254. ISSN: 0362-5915 (cited on page 17).

Conte, D. et al. (2004)
"Thirty Years Of Graph Matching In Pattern Recognition." In: *International Journal of Pattern Recognition and Artificial Intelligence* 18.03, pp. 265–298 (cited on pages 78, 79).

Cook, Stephen A. (1971)
"The Complexity of Theorem-proving Procedures." In: *Proceedings of the Third Annual ACM Symposium on Theory of Computing*. New York, NY, USA: ACM, pp. 151–158 (cited on page 78).

Cordts, Marius et al. (2016)
"The Cityscapes Dataset for Semantic Urban Scene Understanding." In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3213–3223 (cited on page 45).

Costes, Benoît et al. (2015)
"An aggregated graph to qualify historical spatial networks using temporal patterns detection." In: *18th AGILE International Conference on Geographic Information Science* (cited on pages 39, 40).

Couclelis, Helen (1992)
"People manipulate objects (but cultivate fields): Beyond the raster-vector debate in GIS." In: *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*. Ed. by A. U. Frank, I. Campari, and U. Formentini. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 65–77. ISBN: 978-3-540-47333-6 (cited on page 19).

Del Mondo, Géraldine et al. (2010)
"A Graph Model for Spatio-temporal Evolution." In: *Journal of Universal Computer Science* 16.11, pp. 1452–1477 (cited on pages 38, 39, 63).

Demiryurek, Ugur et al. (2009)
"Towards Modeling the Traffic Data on Road Networks." In: *Proceedings of the Second International Workshop on Computational Transportation Science*. New York, NY, USA: ACM, pp. 13–18. ISBN: 978-1-60558-861-2 (cited on page 75).

Domingo, Mathieu, Remy Thibaud, and Christophe Claramunt (2013)
"A Graph-based Model for the Representation of Land Spaces." In: *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. New York, NY, USA: ACM, pp. 516–519. ISBN: 978-1-4503-2521-9 (cited on pages 37, 40).

Dragicevic, Suzana and Danielle J. Marceau (2000)
"A fuzzy set approach for modelling time in GIS." In: *International Journal of Geographical Information Science* 14.3, pp. 225–245 (cited on pages 28, 29).

Dutot, Antoine et al. (2007)
"GraphStream: A Tool for bridging the gap between Complex Systems and Dynamic Graphs." In: *4th European Conference on Complex Systems* (cited on page 93).

Egenhofer, M.J. and A.U. Frank (1989)
"Object-oriented modeling in GIS: Inheritance and Propagation." In: *Ninth International Symposium on Computer-Assisted Cartography: Auto-Carto-9*. USA: American Society for Photogrammetry and Remote Sensing, pp. 588–598 (cited on page 29).

Egenhofer, Max J. and Khaled K. Al-Taha (1992)
"Reasoning about gradual changes of topological relationships." In: *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*. Ed. by A. U. Frank, I. Campari, and U. Formentini. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 196–219. ISBN: 978-3-540-47333-6 (cited on page 30).

Egenhofer, Max J and John Herring (1991)
Categorizing binary topological relations between regions, lines, and points in geographic databases. Tech. rep. University of Maine, Orono (cited on pages 20, 49).

Erwig, Martin et al. (1999)
"Spatio-Temporal Data Types: An Approach to Modeling and Querying Moving Objects in Databases." In: *GeoInformatica* 3.3, pp. 269–296. ISSN: 1573-7624 (cited on page 30).

Esser, J. and M. Schreckenberg (1997)
"Microscopic Simulation of Urban Traffic Based on Cellular Automata." In: *International Journal of Modern Physics C* 08.05, pp. 1025–1036 (cited on page 12).

Fan, Wenfei et al. (2011)
"Incremental Graph Pattern Matching." In: *International Conference on Management of Data*. ACM, pp. 925–936. ISBN: 978-1-4503-0661-4 (cited on pages 81, 108).

Frank, Andrew U. (1996)
"Qualitative spatial reasoning: cardinal directions as an example." In: *International Journal of Geographical Information Systems* 10.3, pp. 269–290 (cited on page 22).

Frank, Andrew U. and Sabine Timpf (1994)
"Multiple representations for cartographic objects in a multi-scale tree - An intelligent graphical zoom." In: *Computers & Graphics* 18.6, pp. 823 –829. ISSN: 0097-8493 (cited on pages 54, 56).

Freksa, Christian (1992a)
"Temporal reasoning based on semi-intervals." In: *Artificial Intelligence* 54.1, pp. 199 –227. ISSN: 0004-3702 (cited on page 24).

– (1992b)
"Using orientation information for qualitative spatial reasoning." In: *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*. Ed. by A. U. Frank, I. Campari, and U. Formentini. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 162–178. ISBN: 978-3-540-47333-6 (cited on pages 21, 30).

Galton, Antony (2001)
"A Formal Theory of Objects and Fields." In: *Spatial Information Theory*. Ed. by Daniel R. Montello. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 458–473. ISBN: 978-3-540-45424-3 (cited on page 20).

– (2008)
"Experience and History: Processes and their Relation to Events." In: *Journal of Logic and Computing* 18.3, pp. 323–340. ISSN: 0955-792X (cited on pages 26, 31).

Gazis, Denos C., Robert Herman, and Renfrey B. Potts (1959)
"Car-Following Theory of Steady-State Traffic Flow." In: *Operations Research* 7.4, pp. 499–505 (cited on page 11).

Gazis, Denos C., Robert Herman, and Richard W. Rothery (1961)
"Nonlinear Follow-the-Leader Models of Traffic Flow." In: *Operations Research* 9.4, pp. 545–567 (cited on page 12).

Geiger, Andreas et al. (2013)
"Vision meets robotics: The KITTI dataset." In: *The International Journal of Robotics Research* 32.11, pp. 1231–1237 (cited on page 44).

Giugno, R. and D. Shasha (2002)
"GraphGrep: A fast and universal method for querying graphs." In: *Object recognition supported by user interaction for service robots*. Vol. 2, pp. 112–115 (cited on page 80).

Goodchild, Michael F (1993)
"The state of GIS for Environmental Problem-Solving." In: *Environmental modeling with GIS*. Ed. by M.F. Goodchild, B.O. Parks, and L.T. Steyaert, pp. 8–15 (cited on page 20).

Goodchild, Michael, Robert Haining, and Stephen Wise (1992)
"Integrating GIS and spatial data analysis: problems and possibilities." In: *International Journal of Geographical Information Systems* 6.5, pp. 407–423 (cited on page 17).

Grenon, Pierre and Barry Smith (2004)
"SNAP and SPAN: Towards Dynamic Spatial Ontology." In: *Spatial Cognition & Computation* 4.1, pp. 69–104 (cited on pages 27, 31, 66).

Guesgen, Hans Werner (1989)
Spatial reasoning based on Allen's temporal logic. Tech. rep. International Computer Science Institute Berkeley (cited on page 22).

Hamblin, C. L. (1972)
"Instants and Intervals." In: *The Study of Time: Proceedings of the First Conference of the International Society for the Study of Time Oberwolfach (Black Forest) — West Germany*. Ed. by J. T. Fraser, F. C. Haber, and G. H. Müller. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 324–331. ISBN: 978-3-642-65387-2 (cited on page 25).

Harary, F. and G. Gupta (1997)
"Dynamic graph models." In: *Mathematical and Computer Modelling* 25.7, pp. 79 –87. ISSN: 0895-7177 (cited on page 61).

Helbing, Dirk (2001)
"Traffic and related self-driven many-particle systems." In: *Reviews of Modern Physics* 73 (4), pp. 1067–1141 (cited on page 13).

Helbing, Dirk and Andreas Greiner (1997)
"Modeling and simulation of multilane traffic flow." In: *Physical Review E* 55 (5), pp. 5498–5508 (cited on page 10).

Hendrix, Gary G. (1973)
"Modeling simultaneous actions and continuous processes." In: *Artificial Intelligence* 4.3, pp. 145 –180. ISSN: 0004-3702 (cited on page 24).

Herman, Robert and Ilya Prigogine (1979)
"A Two-Fluid Approach to Town Traffic." In: *Science* 204.4389, pp. 148–151. ISSN: 0036-8075 (cited on page 10).

Hernández, Daniel (1993)
"Maintaining qualitative spatial knowledge." In: *Spatial Information Theory A Theoretical Basis for GIS*. Ed. by Andrew U. Frank and Irene Campari. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 36–53. ISBN: 978-3-540-47966-6 (cited on page 21).

Holme, Petter and Jari Saramäki (2012)
"Temporal networks." In: *Physics Reports* 519.3, pp. 97 –125. ISSN: 0370-1573 (cited on page 61).

Hoogendoorn, S. P. and P. H. L. Bovy (2001a)
"State-of-the-art of vehicular traffic flow modelling." In: *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering* 215.4, pp. 283–303 (cited on page 13).

Hoogendoorn, Serge P. and Piet H. L. Bovy (1998)
"New Estimation Technique for Vehicle-Type-Specific Headway Distributions." In: *Transportation Research Record* 1646.1, pp. 18–28 (cited on page 15).

Hoogendoorn, Serge P. and Piet H.L. Bovy (2000)
"Continuum modeling of multiclass traffic flow." In: *Transportation Research Part B: Methodological* 34.2, pp. 123 –146. ISSN: 0191-2615 (cited on page 11).

– (2001b)
"Platoon-Based Multiclass Modeling of Multilane Traffic Flow." In: *Networks and Spatial Economics* 1.1, pp. 137–166. ISSN: 1572-9427 (cited on page 11).

Hornsby, Kathleen and Max J. Egenhofer (1997)
"Qualitative representation of change." In: *Spatial Information Theory A Theoretical Basis for GIS*. Ed. by Stephen C. Hirtle and Andrew U. Frank. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 15–33 (cited on page 24).

– (2000)
"Identity-based change: a foundation for spatio-temporal knowledge

representation." In: *International Journal of Geographical Information Science* 14.3, pp. 207–224 (cited on page 28).

Hornsby, Kathleen and Max J. Egenhofer (2002)
"Modeling Moving Objects over Multiple Granularities." In: *Annals of Mathematics and Artificial Intelligence* 36.1, pp. 177–194. ISSN: 1573-7470 (cited on page 31).

Hornsby, Kathleen, Max J. Egenhofer, and Patrick Hayes (1999)
"Modeling Cyclic Change." In: *Advances in Conceptual Modeling*. Ed. by Peter P. Chen et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 98–109. ISBN: 978-3-540-48054-9 (cited on page 27).

Jensen, Christian S., Michael D. Soo, and Richard T. Snodgrass (1994)
"Unifying temporal data models via a conceptual model." In: *Information Systems* 19.7, pp. 513 –547. ISSN: 0306-4379 (cited on page 17).

Jiang, Bin, Christophe Claramunt, and Björn Klarqvist (2000)
"Integration of space syntax into GIS for modelling urban spaces." In: *International Journal of Applied Earth Observation and Geoinformation* 2.3, pp. 161 –171. ISSN: 0303-2434 (cited on pages 37, 40).

Jiang, Jixiang and Michael Worboys (2009)
"Event-based topology for dynamic planar areal objects." In: *International Journal of Geographical Information Science* 23.1, pp. 33–60 (cited on page 39).

Jindal, Tanvi et al. (2013)
"Spatiotemporal Periodical Pattern Mining in Traffic Data." In: *Proceedings of the 2Nd ACM SIGKDD International Workshop on Urban Computing*. New York, NY, USA: ACM, 11:1–11:8. ISBN: 978-1-4503-2331-4 (cited on page 76).

Kahn, Kenneth and G. Anthony Gorry (1977)
"Mechanizing Temporal Knowledge." In: *Artificial Intelligence* 9.1, pp. 87 –108. ISSN: 0004-3702 (cited on page 25).

Kamran, Shoaib and Olivier Haas (2007)
"A Multilevel Traffic Incidents Detection Approach: Identifying Traffic Patterns and Vehicle Behaviours using real-time GPS data." In: *Proceedings of IEEE Intelligent Vehicles Symposium*, pp. 912–917 (cited on page 55).

Kempe, David, Jon Kleinberg, and Amit Kumar (2002)
"Connectivity and Inference Problems for Temporal Networks." In: *Journal of Computer and System Sciences* 64.4, pp. 820 –842. ISSN: 0022-0000 (cited on pages 63, 64).

Khoshafian, Setrag N. and George P. Copeland (1986)
"Object Identity." In: *ACM SIGPLAN Notices* 21.11, pp. 406–416. ISSN: 0362-1340 (cited on page 16).

Klar, A. and R. Wegener (1998)
"A Hierarchy of Models for Multilane Vehicular Traffic I: Modeling." In: *SIAM Journal on Applied Mathematics* 59.3, pp. 983–1001 (cited on page 14).

Kossinets, Gueorgi, Jon Kleinberg, and Duncan Watts (2008)
"The Structure of Information Pathways in a Social Communication Network." In: *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: ACM, pp. 435–443. ISBN: 978-1-60558-193-4 (cited on page 64).

Kovanen, Lauri et al. (2011)
"Temporal motifs in time-dependent networks." In: *Journal of Statistical Mechanics: Theory and Experiment* 2011.11, P11005 (cited on pages 80, 81).

Kowalski, Robert and Marek Sergot (1986)
"A logic-based calculus of events." In: *New Generation Computing* 4.1, pp. 67–95. ISSN: 1882-7055 (cited on page 24).

Langran, Gail and Nicholas R Chrisman (1988)
"A Framework For Temporal Geographic Information." In: *Cartographica: The International Journal for Geographic Information and Geovisualization* 25.3, pp. 1–14 (cited on pages 17, 28).

Laval, Jorge A. and Ludovic Leclercq (2008)
"Microscopic modeling of the relaxation phenomenon using a macroscopic lane-changing model." In: *Transportation Research Part B: Methodological* 42.6, pp. 511 –522. ISSN: 0191-2615 (cited on page 14).

Lighthill, Michael James and Gerald Beresford Whitham (1955)
"On kinematic waves II. A theory of traffic flow on long crowded roads." In: *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* 229.1178, pp. 317–345 (cited on pages 10, 14).

Lin, Yunqing (1991)
"Two theories of time." In: *Journal of Applied Non-Classical Logics* 1.1, pp. 37–63 (cited on page 25).

Maerivoet, Sven and Bart De Moor (2005)
"Cellular automata models of road traffic." In: *Physics Reports* 419.1, pp. 1 –64. ISSN: 0370-1573 (cited on pages 13, 35).

McCarthy, John and Patrick J. Hayes (1969)
"Some Philosophical Problems from the Standpoint of Artificial Intelligence." In: *Machine Intelligence 4*. Ed. by B. Meltzer and D. Michie. reprinted in McC90. Edinburgh University Press, pp. 463–502 (cited on pages 24, 25).

McDermott, Drew (1982)
"A Temporal Logic for Reasoning About Processes and Plans." In: *Cognitive Science* 6.2, pp. 101–155 (cited on pages 25, 26, 66).

Nagel, Kai (1998)
"From Particle Hopping Models to Traffic Flow Theory." In: *Transportation Research Record* 1644.1, pp. 1–9 (cited on page 14).

Nagel, Kai and Michael Schreckenberg (1992)
"A cellular automaton model for freeway traffic." In: *Journal de Physique I* 2.12, pp. 2221–2229 (cited on pages 12, 13).

Nagel, Kai et al. (1998)
"Two-lane traffic rules for cellular automata: A systematic approach." In: *Physical Review E* 58 (2), pp. 1425–1437 (cited on page 12).

Newell, G. F. (1961)
"Nonlinear Effects in the Dynamics of Car Following." In: *Operations Research* 9.2, pp. 209–229 (cited on pages 13, 14).

Oh, S., Y. Kim, and J. Hong (2015)
"Urban Traffic Flow Prediction System Using a Multifactor Pattern Recognition Model." In: *IEEE Transactions on Intelligent Transportation Systems* 16.5, pp. 2744–2755 (cited on page 76).

Papageorgiou, Markos (1998)
"Some remarks on macroscopic traffic flow modelling." In: *Transportation Research Part A: Policy and Practice* 32.5, pp. 323 –329. ISSN: 0965-8564 (cited on page 14).

Paveri-Fontana, S.L. (1975)
"On Boltzmann-like treatments for traffic flow: A critical review of the basic model and an alternative proposal for dilute traffic analysis." In: *Transportation Research* 9.4, pp. 225 –235. ISSN: 0041-1647 (cited on pages 10, 11, 15).

Payne, Harold J (1971)
"Model of freeway traffic and control." In: *Mathematical Model of Public System* 1.1, pp. 51–61 (cited on pages 13, 14).

Peuquet, Donna J (1984)
"A Conceptual Framework and Comparison of Spatial Data Models." In: *Cartographica: The International Journal for Geographic Information and Geovisualization* 21.4, pp. 66–113 (cited on page 17).

Peuquet, Donna J. (1988)
"Representations of Geographic Space: Toward a Conceptual Synthesis." In: *Annals of the Association of American Geographers* 78.3, pp. 375–394 (cited on page 19).

– (1994)
"It's About Time: A Conceptual Framework for the Representation

of Temporal Dynamics in Geographic Information Systems." In: *Annals of the Association of American Geographers* 84.3, pp. 441–461 (cited on pages 18, 19).

Peuquet, Donna J. and Niu Duan (1995)
"An event-based spatiotemporal data model (ESTDM) for temporal analysis of geographical data." In: *International Journal of Geographical Information Systems* 9.1, pp. 7–24 (cited on page 31).

Pipes, Louis A. (1953)
"An Operational Analysis of Traffic Dynamics." In: *Journal of Applied Physics* 24.3, pp. 274–281 (cited on pages 11, 13).

Porta, Sergio, Paolo Crucitti, and Vito Latora (2006)
"The Network Analysis of Urban Streets: A Primal Approach." In: *Environment and Planning B: Planning and Design* 33.5, pp. 705–725 (cited on pages 37, 50).

Prasad Sistla, A. et al. (1997)
"Modeling and querying moving objects." In: *Proceedings of 13th International Conference on Data Engineering*. IEEE, pp. 422–432 (cited on page 30).

Prigogine, Ilya and Robert Herman (1971)
Kinetic Theory of Vehicular Traffic. Elsevier, N.Y. (cited on page 10).

Randell, David A., Zhan Cui, and Anthony G. Cohn (1992)
"A Spatial Logic Based on Regions and Connection." In: *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., pp. 165–176. ISBN: 1-55860-262-3 (cited on pages 20, 48).

Redmond, Ursula and Pádraig Cunningham (2016)
Subgraph Isomorphism in Temporal Networks (cited on pages 80, 81).

Renz, Jochen (2001)
"A Spatial Odyssey of the Interval Algebra: 1. Directed Intervals." In: *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 1*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., pp. 51–56. ISBN: 1-55860-812-5, 978-1-558-60812-2 (cited on pages 23, 48).

Rickert, M. et al. (1996)
"Two lane traffic simulations using cellular automata." In: *Physica A: Statistical Mechanics and its Applications* 231.4, pp. 534 –550. ISSN: 0378-4371 (cited on page 12).

Russell, Bertrand (1903)
Principles of Mathematics. Routledge, U. K. (cited on page 23).

Shoham, Yoav (1985)
"Ten requirements for a theory of change." In: *New Generation Computing* 3.4, pp. 467–477 (cited on page 24).

Siabato, Willington et al. (2018)
"A Survey of Modelling Trends in Temporal GIS." In: *ACM Computing Surveys* 51.2, 30:1–30:41. ISSN: 0360-0300 (cited on page 27).

Sinton, David (1978)
"The inherent structure of information as a constraint to analysis: Mapped thematic data as a case study." In: *Harvard papers on Geographic Information Systems* (cited on pages 17, 18, 35).

Snodgrass, Richard and Ilsoo Ahn (1986)
"Temporal Databases." In: *Computer* 19.9, pp. 35–42. ISSN: 0018-9162 (cited on page 17).

Solnon, Christine (2010)
"AllDifferent-based filtering for subgraph isomorphism." In: *Artificial Intelligence* 174.12, pp. 850 –864. ISSN: 0004-3702 (cited on page 79).

Spéry, Laurent, Christophe Claramunt, and Thérèse Libourel (2001)
"A Spatio-Temporal Model for the Manipulation of Lineage Metadata." In: *GeoInformatica* 5.1, pp. 51–70. ISSN: 1573-7624 (cited on pages 38–40).

Stell, John G. (1999)
"Granulation for Graphs." In: *Spatial Information Theory. Cognitive and Computational Foundations of Geographic Information Science*. Ed. by Christian Freksa and David M. Mark. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 417–432 (cited on page 56).

– (2003)
"Granularity in change over time." In: *Foundations of Geographic Information Science*. Ed. by Matt Duckham, Michael F. Goodchild, and Michael Worboys. London: Taylor and Francis, pp. 95–115. ISBN: 9780429214479 (cited on pages 38, 39).

Strigel, E. et al. (2014)
"The Ko-PER intersection laserscanner and video dataset." In: *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. IEEE, pp. 1900–1901 (cited on page 45).

Tang, John et al. (2010)
"Analysing Information Flows and Key Mediators Through Temporal Centrality Metrics." In: *Proceedings of the 3rd Workshop on Social Network Systems*. New York, NY, USA: ACM, 3:1–3:6. ISBN: 978-1-4503-0080-3 (cited on pages 62, 64).

Treiber, Martin, Ansgar Hennecke, and Dirk Helbing (2000)
"Congested traffic states in empirical observations and microscopic

simulations." In: *Physical Review E* 62 (2), pp. 1805–1824 (cited on pages 13, 35).

Tsichritzis, D. C. and F. H. Lochovsky (1977)
Data base management systems. Academic Press, New York (cited on page 16).

Ullmann, J. R. (1976)
"An Algorithm for Subgraph Isomorphism." In: *Journal of ACM* 23.1, pp. 31–42. ISSN: 0004-5411 (cited on page 79).

Vaqar, Sayyid A. and Otman Basir (2009)
"Traffic pattern detection in a partially deployed vehicular Ad Hoc network of vehicles." In: *IEEE Wireless Communications* 16.6, pp. 40–46 (cited on page 76).

Vento, Mario (2015)
"A long trip in the charming world of graphs for Pattern Recognition." In: *Pattern Recognition* 48.2, pp. 291 –301. ISSN: 0031-3203 (cited on page 78).

Vila, Lluís (1994)
"A Survey on Temporal Reasoning in Artificial Intelligence." In: *AI Communications* 7.1, pp. 4–28. ISSN: 0921-7126 (cited on pages 26, 65).

Vilain, Marc B. (1982)
"A System for Reasoning About Time." In: *Proceedings of the Second AAAI Conference on Artificial Intelligence*. AAAI'82. AAAI Press, pp. 197–201 (cited on page 25).

Wachowicz, Monica and Richard G Healey (1994)
"Towards temporality in GIS." In: *Innovations in GIS 1: Selected Papers from the First National Conference on GIS Research UK*, pp. 105–115 (cited on page 30).

Wardrop, J. G. (1952)
"Road Paper. Some Theoretical Aspects Of Road Traffic Research." In: *Proceedings of the Institution of Civil Engineers* 1.3, pp. 325–362 (cited on pages 9, 35).

Weghe, Nico Van de, Anthony G. Cohn, and Philippe De Maeyer (2004)
"A Qualitative Representation of Trajectory Pairs." In: *Proceedings of the 16th European Conference on Artificial Intelligence*. Amsterdam, The Netherlands, The Netherlands: IOS Press, pp. 1101–1102. ISBN: 978-1-58603-452-8 (cited on page 23).

Wehmuth, Klaus, Artur Ziviani, and Eric Fleury (2015)
"A unifying model for representing time-varying graphs." In: *IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 1–10 (cited on pages 63, 65).

Weiming Hu et al. (2006)
"A system for learning statistical motion patterns." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28.9, pp. 1450–1464 (cited on page 76).

Wong, G.C.K and S.C Wong (2002)
"A multi-class traffic flow model – an extension of LWR model with heterogeneous drivers." In: *Transportation Research Part A: Policy and Practice* 36.9, pp. 827 –841. ISSN: 0965-8564 (cited on page 11).

Worboys, M. F. (1994)
"A Unified Model for Spatial and Temporal Information." In: *The Computer Journal* 37.1, pp. 26–34. ISSN: 0010-4620 (cited on page 29).

Worboys, Michael (2005)
"Event-oriented approaches to geographic phenomena." In: *International Journal of Geographical Information Science* 19.1, pp. 1–28 (cited on pages 26, 32, 39).

Xuan, B. Bui, A. Ferreira, and A. Jarry (2003)
"Computing shortest, fastest, and foremost journeys in dynamic networks." In: *International Journal of Foundations of Computer Science* 14.02, pp. 267–285 (cited on pages 62, 63).

Yuan, May (2001)
"Representing Complex Geographic Phenomena in GIS." In: *Cartography and Geographic Information Science* 28.2, pp. 83–96 (cited on pages 32, 33).

Yuan, May and Kathleen Stewart Hornsby (2008)
Computation and Visualization for Understanding Dynamics in Geographic Domains: A research agenda. Boca Raton, USA: CRC Press (cited on page 28).

Zhou, Yang, Hong Cheng, and Jeffrey Xu Yu (2009)
"Graph Clustering Based on Structural/Attribute Similarities." In: *Proceedings of VLDB Endowment* 2.1, pp. 718–729. ISSN: 2150-8097 (cited on page 67).

Zhu, H. et al. (2017)
"Overview of Environment Perception for Intelligent Vehicles." In: *IEEE Transactions on Intelligent Transportation Systems* 18.10, pp. 2584–2601. ISSN: 1524-9050 (cited on page 44).