Traffic Flow Modelling
conceptual model and specific implementations

Michal Tadeusiak

Erasmus Mundus Masters in Complex Systems Science
École Polytechnique, Paris

M2 Project

Project supervisor: Thomas Lacroix, The CoSMo Company

Abstract

The report describes the development of the traffic flow model performed for the CoSMo Company during the six-months long internship. The CoSMo Company specialises in providing a platform for complex systems modelling and simulation. The main goal of the internship was to make first steps towards traffic modelling using their platform. The report presents an outline of the traffic modelling domain providing an insight into the procedure starting from more general transport planning towards the traffic flow modelling.

A selection of models related to different aspects of traffic flow modelling is considered and implemented: route assignment applying user-equilibrium heuristics; the Cell-Transmission model being a member of broad class of first-order models with static density-flow relation; two pointwise intersection models; a mesoscopic model with pseudo-vehicles able to modify their itineraries; the Variable Time Scales approach allowing for significant speedup of simulations.

Having selected the models and taking into consideration the platform specific restriction a conceptual model, playing a role of generic meta-model was designed. The conceptual model allowed for application of various instantiations models and generation of specific simulations demonstrating various aspects of traffic flow modelling.

Contents

1 Introduction

This report describes the six-month long internship taking place from January till July 2014 in the CoSMo Company. The company was set up in 2010 and provides a specialised platform for modelling and simulation of complex systems. Two main areas of the platform functionality being under development are related to the biology and cities. The CoSMo's goal is to enable of the integrated modelling and simulation of various aspects of the city development, such as economy, ecology, sociology and transport within one platform. For more insight into the companies platform and its applications see the abstract attached to the report. The abstract was written during the internship and was submitted for the symposium: "Toward Integrated Modelling Of Urban Systems"[1] taking place in Lyon, October 2014 (accepted).

The main goal of the internship was to make first steps towards traffic modelling using the CoSMo platform. The first step was to present a concise outline of the traffic modelling domain providing an insight into the entire procedure, from transport planning to traffic flow modelling, allowing to select and implement exemplary, possibly generic models. As a result, the project does not focus on a thorough study of a single model, but instead aims to apply various sub-models covering possibly broad spectrum of traffic modelling problems. Each of the described models is connected with another by either extending its functionality or improving some drawbacks. Therefore the selection of models can be logically inferred.

The remainder of this report is organised as follows. The next section describes the entire procedure of traffic modelling. Then, each of applied sub-models is shortly presented, specifically in the relation with other models. Every model is accompanied by a more detailed description and a demonstration aiming to facilitate their understanding and provide some intuition, but also to check the correctness of the implementation. These sections are attached as dedicated appendices (it is worth at least to get familiar with the demonstration sections).

The models description is followed by a more technical outline of the CoSMo platform, as it had an important impact on the development of the models and simulations.

Finally, getting acquainted with particular features of selected models and the platform specifics the design of the conceptual model, being a main development of the project is explained.

1.1 Traffic Models

Vehicular Traffic modelling[2] is a broad topic covering various aspects such as: transportation planning, traffic flows, traffic control - models describing a constant movement of vehicles, cyclists, pedestrians in different time and space scales (Treiber, 2013). Although they address the problems from different perspectives and focus on different approaches they are all related to each other.

It is worth to emphasise a distinction between the *traffic flow modelling* and *transportation planning* as these two subjects belong to the broader field of *traffic modelling* and can be easily confused due to their close relation.

Following Treiber one can distinguish between the models families based on three aspects: temporal (see Fig. 1), objective (transportation planning investigates relation between the infrastructure and demand, while in traffic flow modelling infrastructure is fixed and demand given externally), subjective (traffic flow modelling is oriented towards drivers behaviour on the road, while transportation planning considers higher level decisions, i.e. choice of activities, destinations, modes of transport and routes).

This project is devoted mainly to traffic flow dynamics, however it was also necessary to work on the traffic assignment as it can be seen as an interface between the transportation planning outputs and traffic flow models' inputs. This is the subject of the following sections.

1.2 Four step model

The most common approach to personal travel modelling which is a starting point to traffic flow modelling is referred to as the Four Step Model (FSM) (McNally, 2008). The steps are as follows:

[1] `http://urbanmodelling.sciencesconf.org/`

[2] The world "vehicular" is significant as there exists a broad field related to the internet traffic of data

Time scale	Field	Models	Aspect of traffic (examples)
<0.1 s	Vehicle dynamics	Sub-microscopic	Control of engine and brakes
1 s			Reaction time, time gap
10 s	Traffic flow dynamics	Car-following models	Acceleration and deceleration
1 min		Macroscopic models	Cycle period of traffic lights
10 min			Stop-and-go waves
1 h			Peak hour
1 day		Route assignment traffic demand	Daily demand pattern
1 year	Transportation planning		Building/changing infrastructure
5 years		Statistics age pyramid	Socioeconomic structure
50 years			Demographic change

Figure 1: Delimitation of traffic flow dynamics from vehicular dynamics and transportation planning (Source: Treiber, 2013)

1. trip generation: an analysed region is first split into zones. Then each of them is assigned with the population and employment distributions based on the land use statistics and forecasts. It renders trips frequencies represented as production (at origins) and attraction (of destinations) measures;

2. trip distribution: a set of individual trips is distributed between the origins and destinations yielding origin-destination (OD) tables of trip demands;

3. mode choice: the trips are split into alternative modes of transport, e.g. trains, buses, cars resulting in factored trip tables reflecting observed travel proportions;

4. route choice: finally, specific routes are selected based on the obtained OD tables.

The last step is commonly referred to as the *traffic assignment*.

1.3 Traffic Assignment

The traffic assignment is based on the principle that road users are trying to find a route between their origin and destination of the shortest travel time. However, the travel times are not fixed attributes of the roads on the way, but depend on the total traffic flows through them. It is therefore not clear a priori which path through the network has the shortest travel time. One of approaches to assign the traffic is to allow these two mechanisms, travel decisions and resulting congestion, compete until a stable flow pattern throughout the road network establishes.

The transport network equilibrium analysis is a very broad topic (Sheffi, 1985). There exist two main principles developed by Wardrop (1952) and resulting from them two optimisation methods: user-equilibrium and system optimum.

User-equilibrium (UE) approach aims to satisfy the Wardrop's first principle stating that: *the journey times on all the routes actually used are equal, and less than those which would be experienced by a single vehicle on any unused route.*

System optimum (SO) approach is based on the Wardrop's second principle and ensures that at equilibrium the average journey time is minimal. The resulting flow pattern would not in general represent an equilibrium situation. After Sheffi, that pattern "can result only from joint decisions by all motorists to act so as to minimize the total system travel time rather than their own". The system optimum could be presumably achieved by imposing dynamic congestion pricing (Ran & Boyce, 1994; Palma et al., 2005), however this would require a more centralised decision making.

Finally the first approach was selected for implementation as it better fits the complex system philosophy of autonomous agents not having an insight into system's optimal configuration.

There exist several algorithm enabling for searching for user equilibrium configuration of flows. In the project, two popular heuristics were implemented, namely: capacity restraint and incremental assignment. They return a selection of routes between origin-destination pairs. A more detailed description of both methods is provided in appendix A, together with a comparison between their performances.

1.4 Traffic flow models

We can divide the traffic flow models into distinct classes depending on the mathematical formulas they are derived from and in general a level of details they provide. Following Treiber (2013) one can distinguish between:

Macroscopic models Macroscopic models treat the traffic flow analogously to gases or liquids. They must satisfy the continuity equation binding the density changes with gradients of the traffic flow:

$$\frac{\partial \rho}{\partial t} + \frac{\partial Q}{\partial x} = 0 \tag{1}$$

The traffic is described by locally aggregated quantities such as densities ρ, flows Q, mean speeds. The models are not able to capture the behaviour of single vehicles which has important consequences, for instance, it is not possible to model a dynamic respond of the flows to the road conditions i.e. rerouteing to the alternative passage, as the flows by itself do not keep the notion of origin and destination. In general any changes to the conditions and therefore to the local flows may call for an expensive recalculation of the flows in the entire network aiming to establish a new equilibrium.

However, thanks to their low level of precision the macroscopic models are considerably faster. Also, as they do not provide many details, they do not need precise settings and deal well with lacks in the input data.

Microscopic models Microscopic models operates on the level of single vehicles, therefore enables for heterogeneous and much more detailed description including the individual positions, velocities, accelerations. The emerging dynamics is an outcome of interactions between individual agents. The behaviour of a car depends on the behaviour of its neighbours. In order to resemble human behaviour the interactions are often given by complicated mathematical formula which has a negative impact on the simulations' performance.

Although the microscopic models can provide valuable details useful, among others, for visualisation purposes, they also call for very precise settings with respect to model parameters and environmental features e.g number of lines, physical representation of intersections.

One of major advantages is that each vehicle knows its origin and destination and can autonomously choose the best route between them. The route can be modified afterwards in the on-line manner as a response to condition changes such as congestion.

1.4.1 Micro/macro models

There is then a tradeoff between the level of detail and the performance. However, there exist some approaches aiming to overcome the disadvantages of the two model types. There are two particularly useful among them: hybrid and mesoscopic models.

Hybrid models Hybrid models, able to combine both approaches make it possible to treat different links of a road network with different levels of detail. By switching from more general macroscopic to the microscopic view one can simulate a zone of particular interest with increased precision while still saving resources in the outer areas. Such approaches are recently under extensive investigation (Leclercq, 2007)

The hybrid models take advantages of the macroscopic and microscopic models in explicit way by applying them in separated areas.

Mesoscopic models In contrast, the mesoscopic models share the advantages of two models by partially merging them.

The traffic flow is described by the aggregated quantities, as in the macroscopic models, however there exist a notion of vehicles - specific for the microscopic formalism. The vehicles are partly autonomous. They do not interact with each other, but all behave according to the local circumstances i.e. they keep the mean speed at the road they are occupying. Particularly, a number of cars can share the same position as their location is used only to indicate whether the vehicle is still within the road dimensions (otherwise it should be transferred to the successive section).

At the same time, each car stores the information about its origin, destination and the optimal route between the two. This allows for effective respond to the conditions on the roads such as obstacles and congestion by dynamic rerouteing.

In addition the described pseudo-vehicles may carry an arbitrary assigned amount of traffic and therefore represent a multiplicity of cars. By reducing the number of vehicles by making them “heavier” it is possible to drastically improve performance of a simulation while still keeping the outcomes valid.

During the internship, two kinds of models were managed to be implemented: the Cell-Transmission Model belonging to the macroscopic class and a mesoscopic model.

2 Sub-models

2.1 Cell-Transmission model

The Cell-Transmission model is an example of a macroscopic model. A road network is divided into homogeneous sections. The flows between them are managed by applying the supply-demand method, i.e. the flows between the neighbouring cells are limited by either demand of the upstream road or supply of the downstream road. Having found agreed inflows and outflows, the density on the road section is inferred using Godunov’s scheme. Finally the total flows throughout the cell is obtained based on the fundamental diagram (a static relation between the density and the total flow).

The detailed description of the model together with an example simulation are provided in the appendix B.

Cell-Transmission model is among the most commonly used models thanks to its generality and simplicity. A proper implementation of the model assures that any model from a broad class of first-order models with static density-flow relation (called Lighthill–Whitham–Richards models) can be applied using the same design, i.e. a structure including a notion of a connector (see the conceptual model 4). A connector links neighbouring roads and based on their demand and supply agrees upon the flow between them.

However, a single connector associates exactly two sections, in a case of a connection between many roads when the interflows have to be agreed collectively a generalised mechanism have to be applied. This problem is addressed in the section 2.2 discussing models of intersections.

There exists also a considerable drawback, related to the problem of convergence. Macroscopic models provide numerical solutions of the continuity equation and in order to converge have to satisfy some restrictions relating the time and space steps used while integrating. An equivalent of the space step Δx is, in the case of CTM, the road’s length. After Treiber, “exact solutions [to the continuity equation] exist if we make sure that neither information (carried by the vehicles or by the propagation velocities) propagates over more than one cell during one time step”.The condition is know under the names of Courant, Friedrichs and Lévy (CFL) (see Appendix E) and binds the time step duration Δt with the road’s length Δx. This causes performance problems which are addressed in section 2.3.

2.2 Intersection

In case of an intersection when a road section may have many sources and destinations there is a need for a generalisation of the supply-demand method. The problem of intersection modelling in the context of macroscopic traffic flow models is relatively poorly investigated. As Lebacque (2005) notices “link boundary conditions must be combined at intersections in order to yield intersection models", however "mathematical textbooks on systems of conservation equations usually skip the subject of boundary conditions".

Based on the supply-demand method two mathematical models were developed. They belong to the class of the pointwise intersection models. They compute the flows between the roads satisfying the conditions of the continuity of flows:

- an inflow to any road does not exceed its supply: $Q_i^{in} \leq S_i$;
- an outflow from any road does not exceed its demand: $Q_i^{out} \leq D_i$;
- the total outflow from the intersection equals the total inflow into the intersection: $\sum_i Q_i^{in} = \sum_i Q_i^{out}$;

The models differ with respect to division of internal flows:

- **Proportional Reduction** model cuts the outflows from a road proportionally to the necessary reduction at the most limiting connection. It can be seen as an intersection with single-line roads. If the vehicles aiming to turn left are blocked due to the flow from opposite direction, entire outflow from the road is reduced;
- **Maximal Throughput** model considers the road outflows' being independent, therefore each outflow is as big as possible given the initial proportions of turning movements in the intersection. This situation would correspond to multiple-line roads.

There are many possible strategies governing the flows, still satisfying the flow continuity conditions (see Holden-Risebro (1995) and Coclite-Piccoli (2005) models)

The constructed algorithms operate on a matrix consisting of every outflow-inflow pairs and based on the roads' demands and supplies calculates the agreed flows. The detailed mathematical descriptions together with examples are provided in appendix C.

2.3 Variable time scales

There exists a limitation called Courant-Friedrichs-Lévy (CFL) condition, mentioned in section 2.1, telling in short that *neither information can propagate over more than one cell during one time step* (Treiber, 2013). This implies:

$$\Delta t < \frac{\Delta x}{V_{limit}} \tag{2}$$

This introduces a significant performance limitation. Given there is a single time step, common for the entire simulation, it has to fulfil the above requirement for the shortest road in the road network. This is not a feasible solution and it calls for optimization.

An answer to this problem proposed recently by Flötteröd and Nagel (2007) is so called *variable time scales* approach where the individual road sections' time step durations are related to the roads' lengths. Shorter roads are updated more frequently, while longer ones only once per several iterations reducing the overall computational costs.

The original article describes the algorithm in the macroscopic regime and then provides some insight into its extension into a mesoscopic model.

Based mainly on the provided description a mesoscopic model was implemented compatible with existing structure of the road network. The model satisfies a general characteristics mentioned in section 1.4.1. In order to be fully operational, the implementation had to be complemented by mechanisms of the creation of vehicles, transfer through the network and removal after reaching the destination. More details regarding the above issues are addressed in appendix D.

The Variable Times Scales approach proves useful already when using a macroscopic model, however a main performance gain is achieved when applying a mesoscopic model. It is due to the fact that in the latter, the state of vehicles (e.g. position) is updated with the same frequency as the road they occupy. The longer is the road, the longer is the time between updates. Moreover, the longer is the road, the more vehicles it can accommodate. Therefore, in the end there is a bigger number of cars invoked relatively less frequently.

The reduction of time step durations yields the neighbouring sections being possibly updated in different times. This call for a buffer mechanism storing the partial information and being able to preserve the communication between faster and slower entities. The latter ones keep their states unchanged between the updates which causes some inaccuracies. There is then a tradeoff between the increased speed of the simulation and the accuracy.

Appendix E is devoted to the Variable Times Scales approach. After explaining the concept it demonstrates its performance with regard to the accuracy and the time demand. In original paper, the method was tested on the road network of Greater Berlin consisting of 2459 links. The longest roads in the network were updated even 64 times less frequently than the shortest ones. While applying the mesoscopic model a simulation speedup was up to factor of 90. In our example of a tiny network consisting of only 5 roads VTS accounted for an 8 times speed-up with the inaccuracy generally not exceeding 1%. For more details refer to section E.2 in appendix E.

The above sections described a selection of specific models devoted to different aspects of traffic flow modelling. The CoSMo platform allows to design a meta-model containing all the above sub-models and being a fundamental structure, upon which instantiation models and simulations can be built and run. The next section will describe the entire platform.

3 The CoSMo platform

The CoSMo platform was designed to suite the specifics of the complex systems. Especially, the CoSMo modelling language (CoSML) allows to define a problem under a systematic regime. There are three main principles constituting the definition of system that can be described using CoSML:

- system is a set of basic entities applying local rules;
- entities can only exchange information by requiring data from their neighbouring entities;
- basic entities can be grouped in compounds that can in turn be considered as entities.

The first step while approaching a complex system is to create a *conceptual model* consistent with above points.

Conceptual model plays a role of a *meta-model.* It captures all the key aspects of the system under scrutiny such as entities, the rules governing their behaviour and the interactions between them.

3.1 Conceptual model

Entity

Entities are the most fundamental elements of any system being modelled. They represent elements constituting the systems i.e. particles, individuals, agents - different names can be used depending on the problem. Each entity is defined by its attributes, by the rules/procedures governing its behaviour and by the way it responds to environmental changes and the stimuli from its neighbours. The entities can be freely grouped into compound entities allowing for implementation of multi-level structures (with no "depth" limitation).

Environment

For better control of the relationships between the entities a notion of environment was introduced. The environment specifies all possible relations between the entities and assures the "locality" of interactions, i.e. only the neighbours can communicate.

A variety of environments is provided, e.g.

- graph, the connections are simply represented by arcs (one-directional) or edges (two-directional). The CoSMo provides many popular network algorithms, such as Dijkstra algorithm;
- room or compartment, allowing for not-restricted communication among all the entities inside;

Furthermore, entities can be part of many environments in the same time allowing for capturing sophisticated multi-level relations.

The structures above allow to describe the most generic representation of the modelled system and should be treated as a meta-model. Based on this the CoSMo engine builder automatically generates a simulation engine that allows to run simulations on specific model instances called *instantiation models*.

3.2 Instantiation model

Instantiation model is a specific application of the conceptual model. Using the predefined entities it creates the population and define the relations between the individuals.

Further it specifies the dynamics by introducing a *scheduler*. The scheduler orders the processes governing the evolution of the entities and the environments by scheduling the actions that each sub-entity instance is supposed to perform.

The simulation can be equipped with *probes* aggregating the data such as internal states of the entities throughout the simulation and transmitting them into *consumers* - providing a user with the output of any form e.g. text, visualisation.

Having been acquainted with basic notions of the CoSMo modelling, we can now focus on details of the developed conceptual model used for traffic flow modelling.

4 Traffic Flow Modelling - Conceptual Model

The conceptual model comprises notions of basic road network elements, already mentioned while describing theoretical models: road sections, connectors, intersections, vehicles.

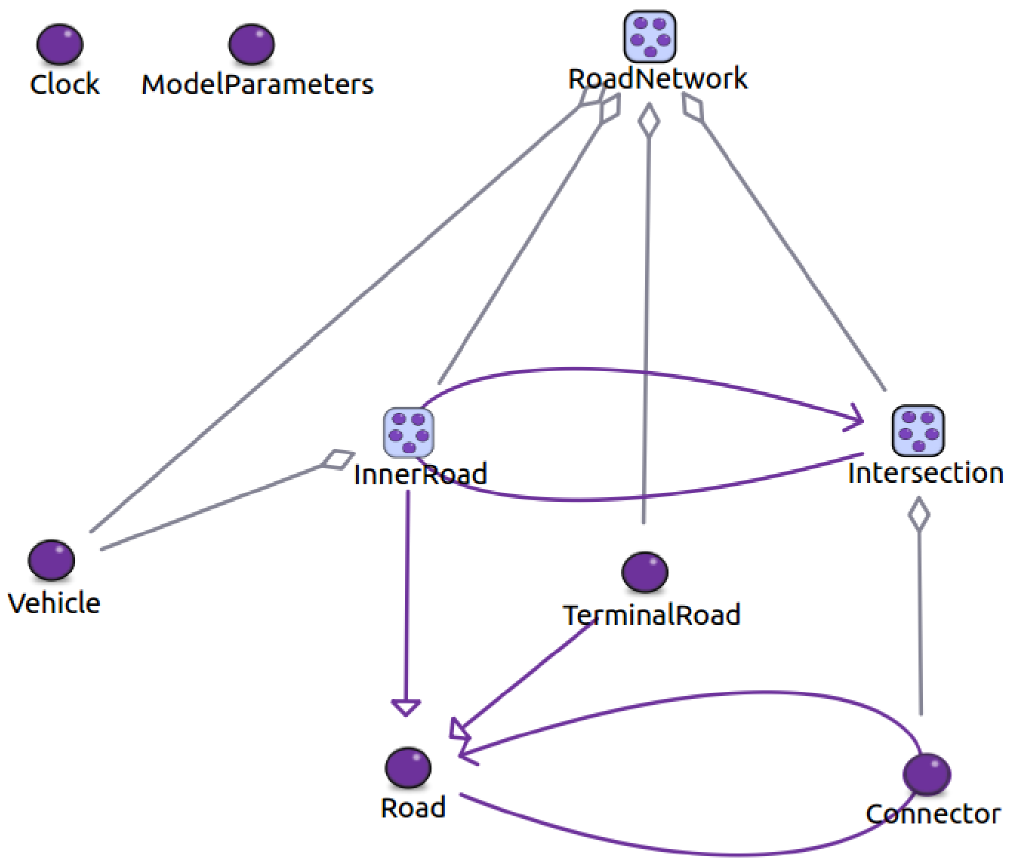


Figure 2: Conceptual model

4.1 Entities

The road network is built based on road sections, intersections and connectors, all interconnected with each other:

RoadNetwork [compound entity] is a root entity in the model. It constitutes the road graph environment consisting of Intersections (nodes) and InnerRoads (arcs, or directed links);

Road [arc entity] is a basic entity representing a homogeneous, one-way section of a road; it stores information such as density, total flow, speed limit (and many other quantities). The only in/out flows may appear at the ends of the section and have to be transmitted through connectors, Based on the current flow they set their demand and supply levels. There are two more specialised entities derived from the *Road*:

- **InnerRoad** [arc compound entity] inner road sections are links between intersections. When using microscopic or mesoscopic models, the road section contains vehicles (therefore it is a compound);
- **TerminalRoad** [basic entity] terminal roads are either origins or destinations of the flow; in macroscopic simulations they keep constant values of supply and demand, in meso-scopic/microscopic they can produce vehicles based on built-in generators;

Connectors [arc entity] are the links between road sections. In simple cases they can autonomously compute the flows between the road. When being part of an intersection they provide the information about the demand and supply which allows the intersection to manage the flows.

Intersection [compound entity] is a compound entity consisting of Connectors and managing the flows between road sections. There cannot exist an intersection without connectors.

The above structure was enforced by following issues:

- due to the supply/demand method the consecutive road sections have to exchange the information about their demand and supply in order to agree on the traffic flow between them. In CoSMo it can be done only by connecting them by an arc or edge, as they have access to their source and destination entities and are able to modify them.
- in most cases intersections consists of many connectors which are potentially interdependent. Therefore there is a need for a collective management of all the connectors in the intersection, accomplished by grouping them in one compound entity, i.e. *Intersection*. Having built a structure providing access to the information about the flow fractions between the attached roads, different mathematical model may be applied, as it was described in section 2.2.
- the shortest path between intersections can be calculated by using Dijkstra algorithm, already built-in in CoSMo. It requires, however, the intersections to be entities, while the road sections to be arcs. They both have to be sub-entities of the *RoadNetwork* constituting a graph.
- vehicles are sub-entities of inner roads and are transferred from one to another. However, as they are autonomous agents and can decide upon their routes, they have to know the entire road network. This is assured by providing them an access to the road network entity and its environment.

Additionally, some additional entities are provided. They can be accessed by any other entity in the conceptual model:

- *ModelParameters* storing general parameters of the models, common for many entities such as: preferred time gap between cars, max density on the roads, effective length of a car (length + safety distance);
- *Clock* measuring the time;

4.2 Instantiation models

Based on the conceptual model, a plethora of various instantiation models can be designed. Then using the CoSMo engine builder, specialised simulations can be generated exercising different settings, focusing on particular aspects and answering specific questions.

It is worth to mention that during the internship a loader compatible with Geographic Information System (GIS) was developed. It is able to generate an instantiation model based on the real geographical data, e.g. the road network of the city of Rennes, France. For details and demonstration refer to appendix F.

5 Conclusion

The final outcome of the internship is a conceptual model allowing for application of various models from a broad spectrum of topics within the Traffic Flow Modelling domain. The submodels were tested and their behaviour demonstrated and analysed qualitatively. The developed structure allows to perform a complete procedure of traffic simulation: loading the geographical data, using the OD-table, assigning the traffic to the network and simulating the resulting flows.

Due to a lack of data the parameters used by different submodels could have not been calibrated for a specific case. However the partial results provide a solid indication to presume that a constructed skeleton is valid and can be successfully applied. A correct cooperation of submodels was already shown, i.e. the test of intersection would not be possible without the compatibility with CTM, the mesoscopic model would not work if the road network structure used by the macroscopic model was not correctly designed, the variable times scales could have been applied because it is compatible with both macroscopic and mesoscopic model.

Finally the conceptual model, consisting of all the above sub-models allow to use them selectively by specialised instantiations models reducing the effort of a modeller when trying to investigate a behaviour of the same system but with different settings. The best examples of this ability are provided in the appendices, in the demonstration sections.

References

[1] Burghout, W. (2004). Hybrid microscopic-mesoscopic traffic simulation.

[2] Coclite, G. M., Garavello, M., & Piccoli, B. (2005). Traffic flow on a road network. SIAM journal on mathematical analysis, 36(6), 1862-1886.

[3] De Palma, A., Kilani, M., & Lindsey, R. (2005). Congestion pricing on a road network: a study using the dynamic equilibrium simulator METROPOLIS. Transportation Research Part A: Policy and Practice, 39(7), 588-611.

[4] Frank, M., & Wolfe, P. (1956). An algorithm for quadratic programming. Naval research logistics quarterly, 3(1-2), 95-110.

[5] Flötteröd, G., & Nagel, K. (2007). High speed combined micro/macro simulation of traffic flow. In IEEE Intelligent Transportation Systems Conference, ITSC (pp. 1114-1119).

[6] Godunov, S. K. (1959). A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. Matematicheskii Sbornik, 89(3), 271-306.

[7] Holden, H., & Risebro, N. H. (1995). A mathematical model of traffic flow on a network of unidirectional roads. SIAM Journal on Mathematical Analysis, 26(4), 999-1017.

[8] Lighthill, M. J., & Whitham, G. B. (1955). On kinematic waves. II. A theory of traffic flow on long crowded roads. Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences, 229(1178), 317-345.

[9] Lebacque, J. P. (2005). Intersection modeling, application to macroscopic network traffic flow models and traffic management. In Traffic and Granular Flow'03 (pp. 261-278). Springer Berlin Heidelberg.

[10] Leclercq, L. (2007). Hybrid approaches to the solutions of the "Lighthill–Whitham–Richards" model. Transportation Research Part B: Methodological, 41(7), 701-709.

[11] McNally, M. G. (2008). The four step model. Center for Activity Systems Analysis.

[12] Ramming, M. S. (2001). Network knowledge and route choice (Doctoral dissertation, Massachusetts Institute of Technology).

[13] Ran, B., & Boyce, D. E. (1994). Dynamic System-Optimal Route Choice and Congestion Pricing. In Dynamic Urban Transportation Network Models (pp. 313-336). Springer Berlin Heidelberg.

[14] Richards, P. I. (1956). Shock waves on the highway. Operations research, 4(1), 42-51.

[15] Sheffi, Y. (1985). Urban transportation networks: equilibrium analysis with mathematical programming methods.

[16] Sparrow, F. T. (1969). The traffic assignment problem for a general network. Journal of Research of the National Bureau of Standards, Series B, 73(2), 91-118.

[17] Sugiyama, Y., Fukui, M., Kikuchi, M., Hasebe, K., Nakayama, A., Nishinari, K., ... & Yukawa, S. (2008). Traffic jams without bottlenecks—experimental evidence for the physical mechanism of the formation of a jam. New Journal of Physics, 10(3), 033001.

[18] Treiber, M., & Kesting, A. (2013). Traffic Flow Dynamics. Traffic Flow Dynamics: Data, Models and Simulation, ISBN 978-3-642-32459-8. Springer-Verlag Berlin Heidelberg, 2013, 1.

[19] U.S. Bureau of Public Roads (1964). Traffic Assignment Manual. U.S. Department of Commerce, Washington, D.C.

[20] Wardrop, J. G. (1952). Some Theoretical Aspects of Road Traffic Research. Proceedings, Institution of Civil Engineers II(1), pp. 325-378.

A complex systems platform for modeling and simulation of cities.

M. Tadeusiak*, M. Edwards, G. Barbaud, T. de Lumley

The CoSMo Company, Lyon

April 2014

Context and motivation

Due to the rapid urbanization in the last decades, more than half of the world's population is currently living in the cities. With the ever increasing complexity of the city processes linking many domains such as economy, ecology, sociology and transport, the city appears increasingly as a system being composed of many dynamically coupled heterogeneous sub-systems requiring the challenge of a revisited, more integrated modeling approach. Accurate city models prove precious to help find and decide upon optimal development choices. For a more holistic understanding of the phenomena observed in the urban areas, the next generation of models has to take into account the variation of temporal and spatial scales of the sub-systems and to provide an efficient way of presenting their coupling and their dynamics. In this context, transportation demand and resulting traffic (at different scales and linked to different fields) play a major role in the city's evolution.

The CoSMo Company is happy to offer a sophisticated tool answering the above needs in their complexity.

Methodology

The CoSMo platform is designed to best suit the complex systems specifics. The CoSMo modeling language (CoSML) enables to design a conceptual model which captures all the key aspects of the complex system being considered, allowing to explain and simulate the underlying phenomenon.

This requires indeed to define entities, which might be associated into groups, at different spatial and conceptual levels, interacting in different ways, following processes with potentially elaborate scheduling rules, leading to the emergence of global phenomena from low-level sub-models.

In CoSMo, systems are modeled as sets of interacting `entities`, each containing specific attributes and evolving following targeted information provided by their neighbors. Such entities can be dynamically created or removed by the model.

Models are not based on global control, all the interactions are local, formalized as the `environment` - a set of relations with a specific structure (i.e a graph or a regular grid) providing a means for entities to communicate. This allows to observe (bottom-up) emergence.

Hierarchical structure can be established by grouping basic entities into one or more compounds, which can be considered as higher level components. This embedding may be indefinitely repeated, providing a way to implement evolving hierarchical multi-scale systems following different axis (geographical, conceptual, ...).

The model's dynamics is managed by a `scheduler`, ordering the processes governing the evolution of the entities and the environments, as well as integrating the influence of the external factors. This scheduler can be dynamically changed by the model following the needs.

Based upon the modeling language, the CoSMo SDK platform has been created. This tool facilitates the modeling task enabling to automatically generate, from a conceptual model, a simulation

*michal.tadeusiak@thecosmocompany.com

engine that allows to run simulations on specific model instances. A simulation can be accompanied by observers keeping record of the selected measures throughout the evolution. Finally, one can use protocols managing the simulations, supporting dynamical modifications of the conditions, helping to study the model, verify its robustness and validate parameter values.

Applications

The CoSMo software was so far successfully applied to the modeling of a broad spectrum of city-related problems, e.g. the EcoCité project focusing on Grand Lyon area, performed together with Veolia and EDF. The models range from yearly scale (city planning), monthly scale (distribution network maintenance), daily scale (supply chain resiliency), down to hourly scale (distribution network crisis management, traffic model).

The Urban Planning model covers various fields: economy, population, transport and ecology. City population growth is driven by life cost and quality, itself being affected by accessibility and environmental constraints. Newcomers increase traffic and have an impact on both ambient (pollution and noise) and accessibility (traffic jams). On top of this, housing availability is driven by an economic model that is influenced by the demand for apartments (supply and demand model). The Urban Planning model operates at the scale of districts and at the time scale of years.

On the contrary, the Traffic Model regulates the flows at a time scale of minutes to hours. Depending on the granularity needed one can perform a macro simulation regarding densities and flows, or a micro simulation focusing on single vehicles. Furthermore, CoSMo permits to simulate with identical protocols (and therefore provide a meaningful comparison between) such two models of the same system but with differing mathematical descriptions corresponding to different levels of spatial precision.

Focusing on urban planning and traffic - two models related to city and transport, we can easily understand the interest to link them together. In such coupling, the Urban Planning model foresees the evolution of the city in the next decades providing the traffic supply and demand distribution. And, while in some bounds, the impact of some small traffic increase can be described by a macro model, getting just out off the bound can make slight traffic bursts over short time periods lead to major traffic jams. In such cases, a more precise traffic model can be used to get a more detailed picture, that can be re-injected into the Urban Planning model providing the information about the transport accessibility.

Conclusion

As the above example models show, CoSMo is a powerful tool to model complex systems such as the city.

Coupling multiple models helps to understand complex systems and to better predict details. A population model itself cannot anticipate the influence of traffic jams and pollution and, modeled alone, a transport model can't be used to forecast future traffic evolution without the information of population repartition.

At the same time, thanks to its modular architecture, CoSMo allows for implementation of multi-scale combined model in one simulation. While simulating one decade of traffic at hour scale is not realistic (nor useful), making a full abstraction of traffic as a transport model can lead to inaccurate results. Switching between scales on key time and spatial elements (when and where the potential traffic jams are) increases performance of a simulation making it faster or more precise where needed.

All these make the CoSMo language and platform an outstanding tool in modeling heterogeneous complex systems in a modular way and efficiently providing details of their dynamics.

A Route assignment

A.1 User-equilibrium algorithms

A general procedure aiming in obtaining the user-equilibrium traffic flow in the road network is based on the following steps:

1. initialisation: travel-times are calculated given the maximum allowed speed;

2. flow update: the shortest-path search using travel-times as weights is performed to select the best routes followed by all-or-nothing traffic assignment;

3. travel-time update: based on the new flows, the travel-times are calculated (this step will be more precisely explained in section A.2);

4. termination: procedure is terminated after reaching an equilibrium (strictly speaking when the differences of flows or times between the successive iterations falls below a predetermined threshold) or after an arbitrary number of repetitions; the final flows are either directly taken from the last iteration or averaged over several of them.

Different algorithms may introduce some alterations to the above steps, however a general idea remains consistent.

Sheffi (1985) in his overview proposed several techniques for solving the user equilibrium problem. He started with introduction of two popular heuristics described below.

Incremental Assignment
This procedure splits the flows between origin-destination pairs into M portions and gradually assign them in all-or-nothing manner. Flow portions are added to the flows at every road the shortest path leads through. Having iterated over every O-D pair and having increased the flows the travel-times are updated. Procedure is finished when all the portions are assigned. It is therefore worth to notice that there is no measure of actual convergence to the equilibrium. A parameter that has an indirect impact on the resulting travel-times is the number of portions M - instinctively the larger it is the closest the times should be to the equilibrium.

Capacity Restraint
The algorithm greatly follows the general procedure, however there exists a popular modification[3] resulting in more smooth convergence. Initially, new travel-times result directly from the actual flows in the links. This may lead to undesirable oscillations of flow assignment from iteration to iteration[4] In order to suppress oscillations the modified version takes into account also the former value:

$$\text{original assignment: } t_i^n = T_i(x_i^{n-1}) \quad | \quad \text{smoothing: } t_i^n = 0.75 \cdot T_i(x_i^{n-1}) + 0.25 \cdot t_i^{n-1} \tag{3}$$

where: t_i^n and x_i^n are a travel-time and a flow through link i at iteration n, respectively; T_i calculates the travel-time based on given flows (more about the T_i function in A.2).

After the assignment is terminated, the final flows are calculated as an average over the last four iterations.

The *capacity restraint* procedure is a special case of more general method applying smoothing with dynamic coefficient. Let the flows at iteration n be given by $\{x_i^n\}$. The next iteration results in auxiliary flows $\{y_i^n\}$. The aforementioned example of flows oscillations suggests that the best flow split lays somewhere between the two vectors:

$$x_i^{n+1} = x_i^n + \alpha_i(y_i^n - x_i^n) \tag{4}$$

[3]adopted among others by the U.S. Federal Highway Administration as part of transportation planning package.

[4]In short: there are two roads between origin and destination, A and B, A slightly shorter. In first step all the traffic is assigned to road A yielding huge congestion. In the next assignment the road B, although longer, will be favoured, leading to the total flow rerouteing yielding, in turn, huge congestion at B. To be continued.

where: $\alpha_i \in [0, 1]$ referred to as *move step* has to be found.

It's worth to notice that in the original *Capacity Restraint* algorithm $\alpha = 1$, while in modified version $\alpha = 0.75$.

In the project the focus was put on providing a possibly generic structure allowing for facilitated implementation of various algorithms. Therefore in the end two heuristic procedures were implemented for comparison. The latter one, as mentioned, can be in future improved by adding a procedure returning optimal α[5].

A.2 The shortest path

The shortest path is found by using the Dijkstra algorithm. Each road section has a weight reflecting its attractiveness. Many different formulas for weights may be proposed, starting from the most basic, considering the length of the road, towards more complicated formulas including for example: current mean speeds, densities, capacities.

There exists a commonly used formula returning a measure of expected travel time based on the current flow on a road versus its capacity. It was developed by the Bureau of Public Roads (BPR) in 1964:

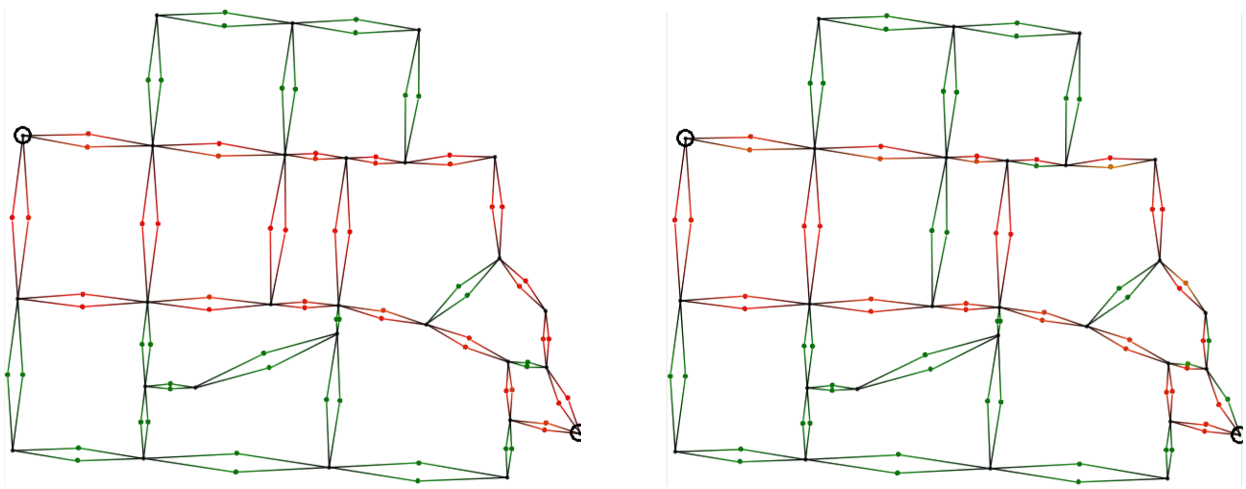
$$\tau_i = t_i \left[1 + 0.15 \left(\frac{Q_i}{C_i} \right)^4 \right] \tag{5}$$

Again, the formula used can be freely modified so to achieve the best possible behaviour corresponding to data (when provided).

A.3 Demonstration

A road network was created consisting of nearly 40 roads. Origin and destination of traffic were fixed in two intersections on opposite sides of the network. Routes were assigned using the two described algorithms. For the comparison to be meaningful, equivalent settings were used, i.e. in case of Incremental Assignment the total flow was split into 10 portions, so the traffic was assigned in 10 iterations. Similarly, 10 iterations were performed while using the Capacity Restraint method.

After finding the routes, the flows were injected into the system. The overall inflow to the network was much bigger then the outflow which resulted in increasing congestion. Fig. 3 presents the outcomes of two procedures, namely the densities on the roads in the saturation state.



(a) Capacity restraint (b) Incremental

Figure 3: Comparison of route assignment algorithms.

[5]A problem of finding an α allowing for fast convergence can be reformulated as line programming problem. There exists a numerical scheme based on the Frank-Wolfe (1956) algorithm proposed by Dafermos and Sparrow (1969) applying the linear search optimising α after each iteration.

Immediate observation is that the both algorithms selected the shortest paths and roads at the peripheries are not occupied. However, the number of alternative routes vary. Capacity restraint method assigned traffic to 5 paths, while Incremental algorithm chose only 4.

In both cases, the goal was to find an equilibrium configuration in which travel times along every route are possibly close to each other. Fig. 4 shows a comparison between travel times.

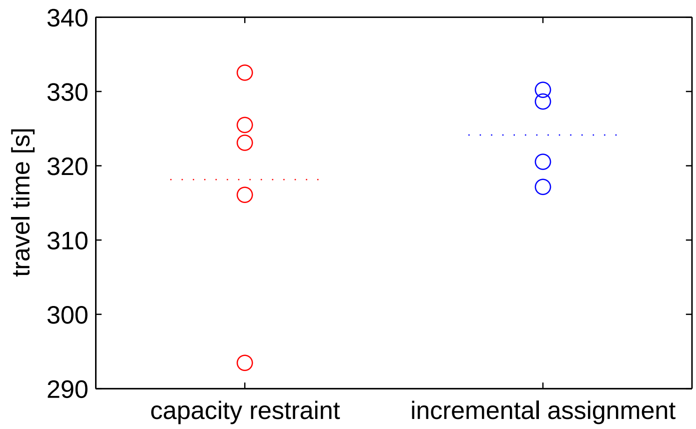


Figure 4: Travel times.

The travel times on particular routes are indicated by circles, the mean times are illustrated as dotted lines. One can see that a bigger number of alternative paths results in shorter travel times on average. However, at the same time relative differences between individual times may grow. Incremental assignment returned smaller number of paths, but properly assigned the flows, so the travel times are similar.

To summarise, in this case, with respect to the user-equilibrium objective, the Incremental Assignment algorithm seems to be working better. However, the resulting travel times are on average worse - it is then always a decision of a modeller, which algorithm to use and which outcomes are closer to the expectations.

B Cell-transmission Model

This appendix describes Cell-transmission Model based on Treiber's book *Traffic Flow Dynamics* (chapter 8) and demonstrates a simple example.

B.1 LWR Models

All macroscopic models are based on the continuity equation binding the rate of change of the density with flow gradients:

$$\frac{\partial \rho}{\partial t} + \frac{\partial Q}{\partial x} = 0 \tag{6}$$

where: $\rho(x,t)$ - density, $V(x,t)$ - speed, $Q(x,t) = \rho(x,t)V(x,t)$ - flow.

There exists an approach proposed by Lighthill, Whitham (1955) and independently Richards (1956) assuming a static, empirically inspired function relating the flow with the density, referred to as the *fundamental diagram*:

$$Q(x,t) = Q_e(\rho(x,t)) \tag{7}$$

This allows to rewrite the continuity equation in a form of first-order PDE, also called, after the names of the authors the LWR model:

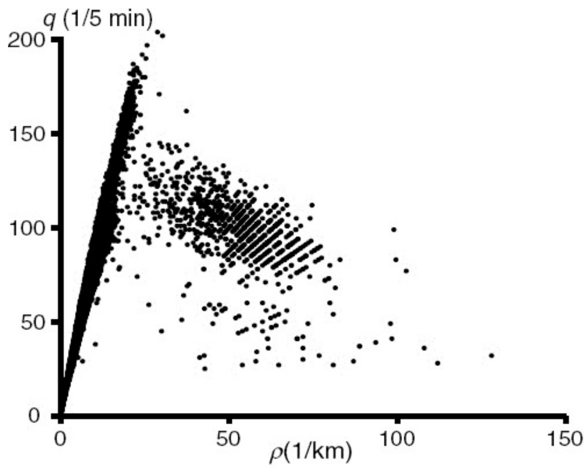
$$\frac{\partial \rho}{\partial t} + \frac{dQ_e(\rho)}{d\rho}\frac{\partial \rho}{\partial x} = 0 \qquad \text{LWR model} \tag{8}$$

The discrete version of the above equation is given by:

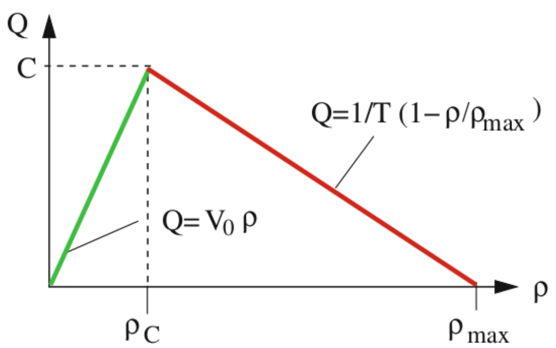
$$\rho_k(t+\Delta t) = \rho_k(t) + \frac{\Delta t}{\Delta x_k}\left(Q_k^{up} - Q_k^{down}\right) \tag{9}$$

Where: Q_k^{up} - inflow, Q_k^{down} - outflow, Δx - road section's length, Δt - time step.

B.1.1 Cell-Transmission Model



(a) Typical fundamental diagram (source: Sugiyama, 2008)



(b) Triangular approximation of a typical fundamental diagram; used in CTM.

Figure 5: Fundamental diagrams (source: Treiber, 2013)

A typical fundamental diagram (Sugiyama, 2008) presenting data being measured on a freeway is shown in Fig. 5a. One can observe a linear dependence of the flow and density followed by a sudden drop around a value of 25 vehicles per kilometre. This value is called *critical density* and corresponds to the maximal flow possible on the road. Larger densities mean shorter distances between cars which can be associated with reduced speed resulting in overall drop of flow. The above empirical relation is usually approximated. Cell-Transmission Model is a simplest LWR model, utilising a triangular approximation showed in Fig. 5b. One can observe free flow (green) and congested (red) sections demarcated by the critical density ρ_c.

B.1.2 Numerical solution

The LWR models are generally solved by dividing roads into homogeneous sections and updating their states in discrete time. In general, an appropriate numerical integration of the LWR models is Godunov's scheme. For the CTM it can be simplified to intuitive "supply-demand" method, where either supply or demand is a flow-limiting factor.

First we define a cell as a homogeneous section of a road, enumerated by k, accompanied by following quantities:

Q^{tot}	current total flow through the cell;
Q^{up}	inflow;
Q^{down}	outflow;
C	capacity (usually constant, can be used as a parameter).

Fig. 6 depicts the relations between neighbouring cells.

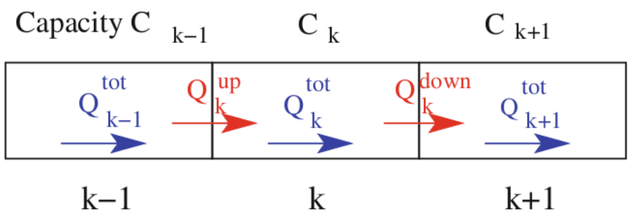


Figure 6: CTM: relations between neighbouring cells (source: Treiber, 2013)

Supply and demand method

The key quantities of the method, *supply* and *demand* can be introduced based on the fundamental diagram. They are both monotonically decreasing functions of the density, limited by the roads capacity C, as it is shown in Fig. 7.

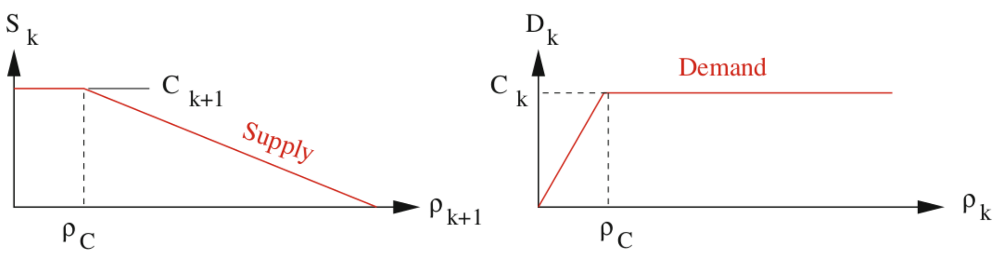


Figure 7: Supply and demand functions. Note a relation with the triangular approximation of fundamental diagram 5b (source: Treiber, 2013)

Therefore the supply and demand are given either by the capacity or by the current traffic flow in a cell, where by cell we describe a homogeneous section of a road (C_k):

Supply : is given by the maximum flow the receiving cell can accommodate:

$$S_k = \begin{cases} Q_k^{tot} & \rho_k > \rho_c, \text{ congested flow,} \\ C_k & \text{otherwise.} \end{cases} \tag{10}$$

Demand : is given by a potential outflow from a cell. The LWR model assumes that the outflow from a congested cell always takes on the maximum value:

$$D_k(t) = \begin{cases} Q_k^{tot} & \rho_k \leq \rho_c, \text{ free flow}, \\ C_k & \text{otherwise}. \end{cases} \tag{11}$$

In / out flows The outflow from the preceding cell must be equal to the inflow into the successive cell and as neither the demand nor the supply can exceed each other, the rule writes simply:

$$Q_k^{up} = Q_{k-1}^{down} = min(S_k, D_{k-1}) \tag{12}$$

Total flow Based on the inflow and outflow the density can be calculated:

$$\rho_k(t + \Delta t) = \rho_k(t) + \frac{\Delta t}{\Delta x_k}\left(Q_k^{up} - Q_k^{down}\right) \tag{13}$$
$$Q_k^{tot}(t + \Delta t) = Q_e\left(\rho_k(t + \Delta t)\right) \tag{14}$$

B.2 Demonstration

An evolution of the traffic on a single road was simulated. The road was split into 3 sections, starting the numeration from the upstream end (see Fig. 8). Initially all the sections were empty. The inflow was grater than the outflow, therefore we expect a traffic jam will appear in the downstream end of the road and its front will be moving upstream as the road gets filled with cars. We should therefore observe a gradual increase of the densities. The saturation should be reached first in the downstream section then in the following sections.

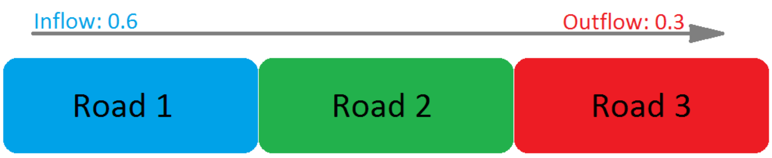


Figure 8: A sequence of 3 roads used in the example.

Fig. 9 presents the resulting evolution, it depicts not only the density but also other relevant quantities, each of them providing valuable insight into the process. The presented results can be interpreted as follows:

- **demand**: before the density exceeds the critical value demand is given by the current flow which, in turn, depends on the density accordingly to the fundamental diagram $Q_e(\rho)$. Given that the $Q_e(\rho)$ is increasing for $\rho < \rho_c$ we should expect an increase of the demand correlated with the increase of the density. The maximum value, related to the road capacity $C \approx 0.56$ is reached when density at the corresponding road arrives at $\rho_c \approx 0.04$, which we indeed observe;

- **supply** in contrary, is given by the roads capacity for $\rho < \rho_c$ and then by the total flow. As the relation $Q_e(\rho)$ is decreasing after reaching the critical density, both the total flow and the supply eventually drop, starting from downstream road which gets filled first;

- **inflows** and **outflows** are coupled, which we can easily observe. The inflow into the upstream road is limited by the capacity (the inflow set in the simulation slightly exceeds it). The inflows into the remaining roads increase in the beginning together with the density and start to decrease when ρ_c is reached. Outflow behaves in analogous way. Eventually all the flows reach a stable value corresponding to the outflow of the downstream end section;

- **total flow** is strictly related to the density by the fundamental diagram $Q_e(\rho)$;

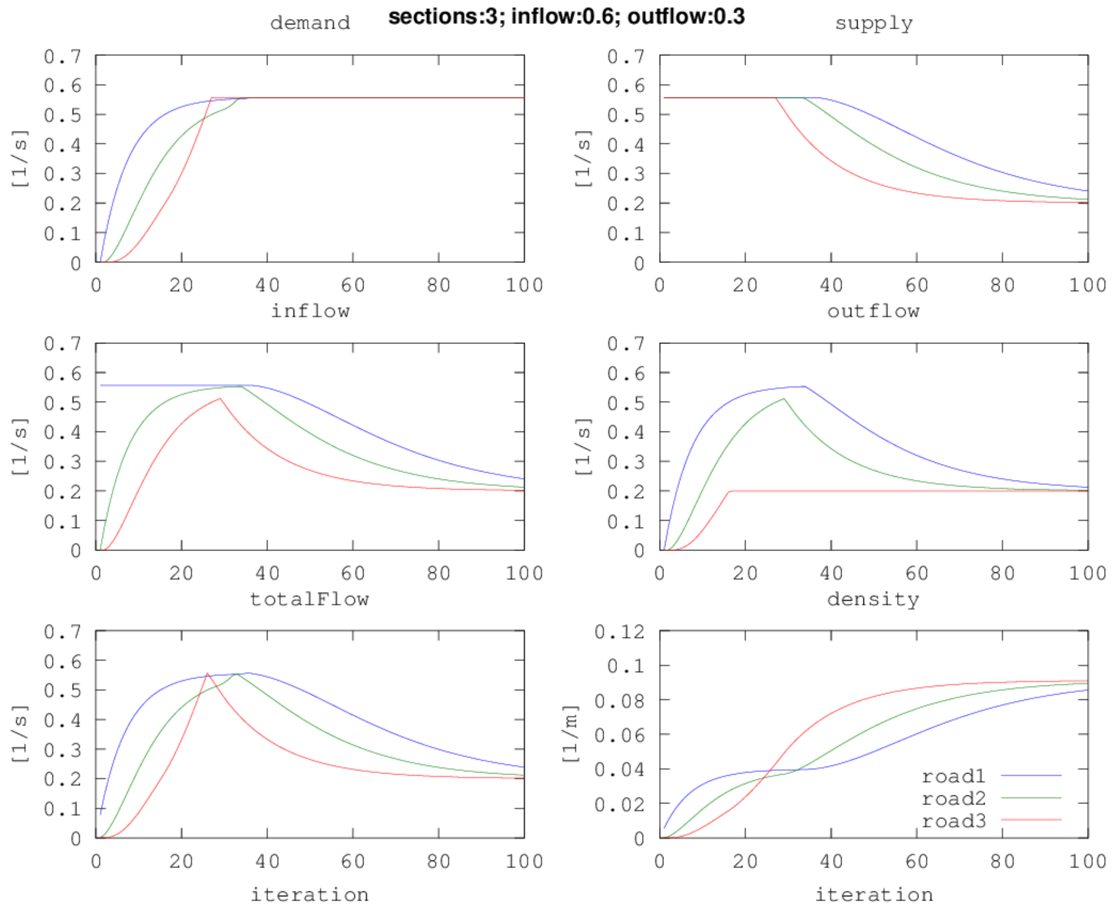


Figure 9: Flow dynamics in the sequence of three road sections.

- finally the **density** behaves, in general, as expected. Although the densities arrive to a limit, it is not equal to the maximum density $\rho_{max} = 0.12$ possible. The maximum is reached only when there is no outflow from the road, otherwise it is smaller. Also, a plateau can be seen, especially visible for "road1", which can be explained as follows:

  1. the first boost of the density is due to the high inflow and lack of outflow (outflow is small as the density is still modest, or we can also say, the cars haven't got yet to the road's end);
  2. the density stabilises when both the inflow and outflow equilibrate;
  3. as the subsequent section gets crowded, the outflow drops and the inflow can be accommodated only by the gradual rise of the density until the limit, followed by a decline of the inflow down to the outflow level.

B.3 Conclusion

The implementation of the CTM model was verified qualitatively, the resulting dynamics meet the expectations. A proper application of the CTM model, means that the designed structure of the model enables for use of any model from the LWR family.

C Intersection

In order to model intersections when any number of inflow/outflow sections may appear the demand-supply method had to be extended. Two models of intersection with N two-way roads is proposed. Each of them is characterized by demand D, supply S, capacity C, etc. There exist *connectors* between any possible pair of the roads' inflow and outflow lines, e.g. arc ϕ_{ij} connects the outflow end of road i with the inflow end of road j, as it is depicted below:

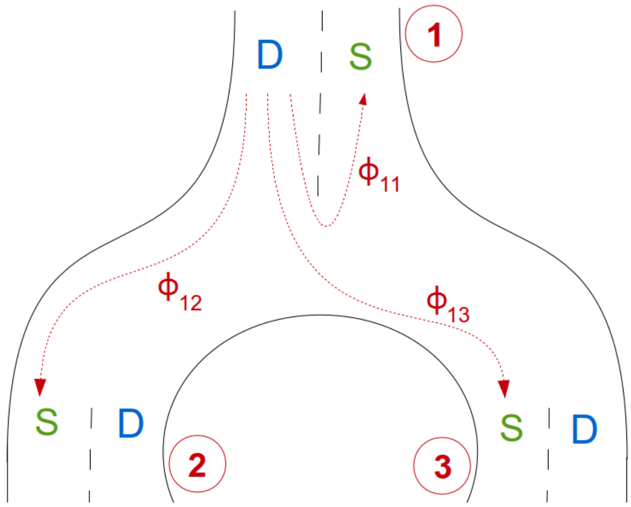


Figure 10: An example of an intersection

For the clarity, the arcs from roads 2,3 are not shown.

Each arc stores an information about the relative flow ϕ, e.g: $\phi_{12} = 0.6$ means that 0.6 of the total outflow from road 1 heads to road 2. It doesn't mean the actual flow from road 1 into road 2 is equal to $D_1\phi_{12}$ - it's only a relative demand that has to be agreed with the supply provided by road 2. First let introduce the notation of common terms:

First we define the demand D and supply S vectors of size equal to the number of roads N:

$$D = \begin{pmatrix} D_1 \\ \vdots \\ D_N \end{pmatrix} \qquad S = (S_1, \cdots, S_N) \tag{15}$$

The relative flow factors constitute the *flow matrix* $\Phi = (\phi_{ij})$ of size $N \times N$. Columns represent roads entering the intersection, while rows are related to the exiting ones. The flow matrix has to satisfy condition:

$$\sum_{j=1}^{N} \phi_{ij} = 1 \tag{16}$$

as all the outflow fractions must sum to 1.

In the next steps some auxiliary values are calculated used further to find final in- and outflows. Similarly to *Matlab* notation, the element-wise operations are indicated by a dot before the operation symbol, i.e. "./" element-wise division, ".∗" element-wise multiplication[6].

downstream demand DD : vector D represents the demand of the upstream road. In order to know the demand for the traffic in the downstream roads, the former one has to be multiplied by the corresponding flow fractions:

$$DD = \Phi \cdot D \tag{17}$$

[6]for precise definitions of the operations refer to Matlab's documentation: `www.mathworks.com/help/matlab/matlab_prog/array-vs-matrix-operations.html`

overflow Γ : downstream demand has to be confronted with supply in order to detect potential overflow:

$$\Gamma = DD \ ./ \ S^T \tag{18}$$

overflow reduced Γ^* : overflows have to be reduced so values don't exceed 1:

$$\Gamma^* = (\gamma_i^*) \ , \quad \gamma_i^* = max(\gamma_i, 1) \tag{19}$$

The above steps are common, the following paragraphs present two algorithms resulting in different flows, but still respecting all the restrictions (see section 2.2)

There is a mathematical issue appearing when a road section does not allow for any inflow ($S_i = 0$). Then, while calculating Γ there is a division by zero. As the product of such operation is in general undefined (DD_i can also be zero), we use a following rule:

Note 1. *Whenever there is division by zero, we set zero as a result.*

This mathematical inconsistency will not cause any problems as any value derived in this way will be always related to the "zero-outflow" road and sooner or later will be multiplied by 0, as seen in the examples below.

C.1 Algorithm I: Proportional Reduction (PR)

This algorithm ensures that the proportions between the partial outflows stay unchanged in case of flow reduction due to insufficient supply. If any of partial outflows is cut by half all the other would be reduced too, even though the corresponding supplies remain big enough. In reality this would correspond to a situation when an outflow in a single direction blocks other outflows, e.g. cars waiting to turn left on an one-line road disturbs entire circulation. The below mathematical procedure returns expected values.

binary matrix Φ^0 : tells between which roads there is any flow:

$$\Phi^0 = (\phi_{ij}^0) \ , \quad \phi_{ij}^0 = \begin{cases} 1 & \text{if } \phi_{ij} \neq 0 \\ 0 & \text{if } \phi_{ij} = 0 \end{cases} \tag{20}$$

overflow matrix M^* : represents individual overflows of every upstream/downstream pair:

$$\Phi^* = \Phi^0 \ .* \ \Gamma^* \tag{21}$$

maximum overflow Γ^{max} : selection of maximum overflows corresponding to every upstream road. In short, if outflow from a road i is limited by insufficient supply of downstream roads, the i^{th} term will be related to the biggest limitation and will be given by a maximum value in each column of Φ^*:

$$\Gamma^{max} = max_{columns}\{\Phi^*\} \tag{22}$$

reduced flows matrix Φ_{PR} : is obtained by dividing terms from the original flow matrix by overflows. The overflow values are equal in each column so the proportions of partial outflows are preserved:

$$\Phi_{PR} = \Phi \ ./ \ \Gamma^{max} \tag{23}$$

Finally, having reduced the relative flow shares, the absolute values can be acquired

absolute flows Q_{PR} : absolute values of traffic exchange between every upstream/downstream pair comes from the reduced flows and original demand:

$$Q_{PR} = \Phi_{PR} \ .* \ D^T \tag{24}$$

absolute inflows Q_{PR}^{in} **and outflows** Q_{PR}^{out} : are appropriate sums retrieved from Q_{PR} and cannot exceed initial supply and demand values:

$$Q_{PR}^{in} = \sum_{in\ rows} Q_{PR} \leq S \qquad\qquad Q_{PR}^{out} = \sum_{in\ columns} Q_{PR} \leq D \tag{25}$$

Section C.3 provides an example illustrating the above calculations.

C.2 Algorithm II: Maximum Throughput

This algorithm ensures that the outflows adapt to demand/supply limitations to reach the maximum throughput possible. In reality this would correspond to situation were the individual outflows are independent, i.e. even though one of turns is blocked, the vehicles pursuing other directions can pass.

The mathematical procedure reads as follows:

reduced flows matrix Φ_{MT} : the flow fractions are reduced based on the overflow fractions regardless the proportions:

$$\Phi_{MT} = \Phi \;./\; \Gamma^* \tag{26}$$

absolute flows Q_{MT} : in analogy to the previous algorithm:

$$Q_{MT} = \Phi_{MT} \;.*\; D^T \tag{27}$$

absolute inflows Q^{in}_{PR} **and outflows** Q^{out}_{PR} : are appropriate sums retrieved from Q_{MT} and cannot exceed initial supply and demand values:

$$Q^{in}_{MT} = \sum_{in\ rows} Q_{MT} \leq S \qquad\qquad Q^{out}_{MT} = \sum_{in\ columns} Q_{MT} \leq D \tag{28}$$

Section C.3 provides an example illustrating the above calculations.

C.3 Example

Let's consider a simple example consisting of 3 roads, see Fig. 11. One of them is a source of traffic, while the remaining two are sources of different supply capabilities. Half of the traffic heads into exit 2, other half into exit 3. As the total demand (0.5) exceeds total supply (0.3), partial flows between the roads will have to be limited depending on downstream roads' supplies and the algorithm used.

In both cases any partial flow cannot exceed $0.5 \times 50\% = 0.25$. We expect the following results:

- PR: the limiting factor is in this case supply of road 3 $S_3 = 0.1$. As the flows split into halves and are reduced proportionally the resulting partial flows should both be equal to S_3;
- MT: capacities of both downstream roads are smaller then 0.25, therefore the algorithm will make a maximum of the available throughput and resulting partial flows will reach the supply values.

Sections below presents all the calculations in detail, so both algorithms can be better understood. Moreover, an approach for dealing with indeterminate mathematical forms is presented.

Common terms

Based on the figure 11:

$$D = \begin{pmatrix} 0.5 \\ 0 \\ 0 \end{pmatrix} \qquad\qquad S = (0, 0.2, 0.1) \tag{29}$$

$$\phi = \begin{pmatrix} \phi_{11} & \phi_{12} & \phi_{13} \\ \phi_{21} & \phi_{22} & \phi_{23} \\ \phi_{31} & \phi_{32} & \phi_{33} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0.5 & 1 & 0 \\ 0.5 & 0 & 1 \end{pmatrix} \tag{30}$$

Values ϕ_{22} and ϕ_{33} are chosen so the sums in columns equal 1, however they are not meaningful as the demands of roads 2 and 3 are zero anyway.

Following the common steps:

$$DD = \Phi \cdot D = \begin{pmatrix} 0 & 0 & 0 \\ 0.5 & 1 & 0 \\ 0.5 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0.5 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0.0 \\ 0.25 \\ 0.25 \end{pmatrix} \tag{31}$$

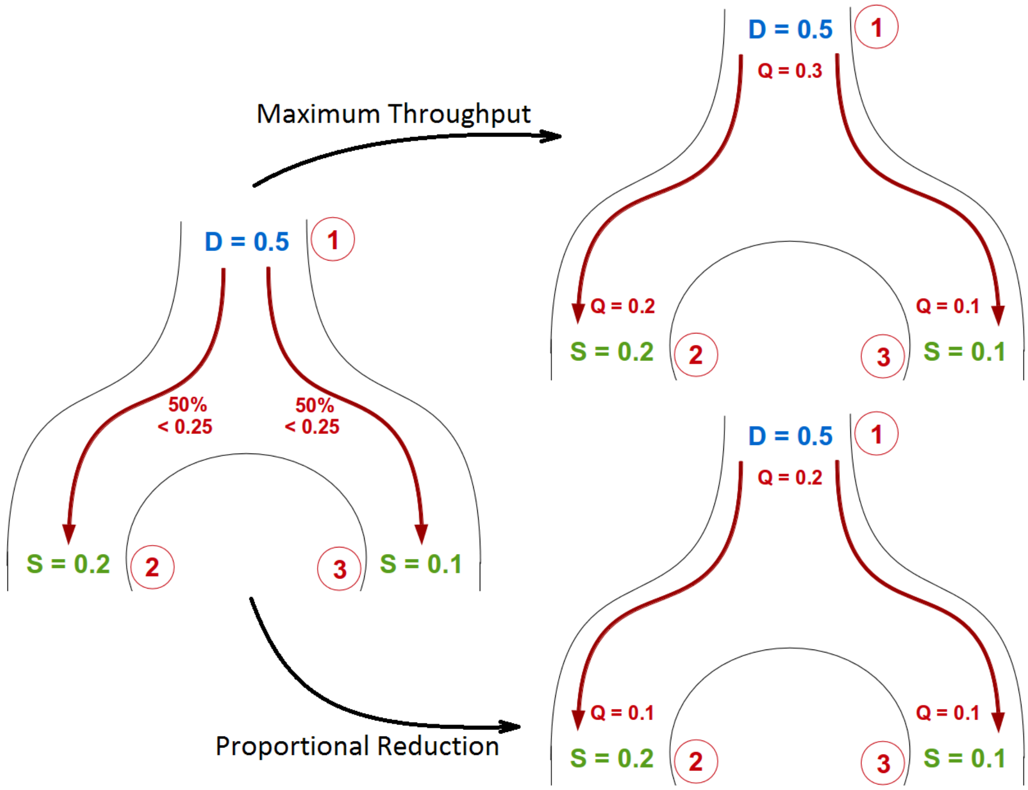


Figure 11: Example of intersection: symmetric diversion.

These are the already mentioned partial flow limits that cannot be exceeded.

It is necessary to point out now, that road 1 offers no supply $S_1 = 0$, which causes some mathematical problems:

$$\Gamma = DD \ ./ \ S^T = \begin{pmatrix} 0 \\ 0.25 \\ 0.25 \end{pmatrix} ./ \begin{pmatrix} 0 \\ 0.2 \\ 0.1 \end{pmatrix} = \begin{pmatrix} 0/0 \\ 0.25/0.2 \\ 0.25/0.1 \end{pmatrix} \overset{\text{note 1}}{=} \begin{pmatrix} 0 \\ 1.25 \\ 2.5 \end{pmatrix} \tag{32}$$

Where the division by zero rule mentioned in note 1 was applied.

$$\Gamma^* = max(\Gamma, 1) = \begin{pmatrix} 1 \\ 1.25 \\ 2.5 \end{pmatrix} \tag{33}$$

Proportional Reduction

$$\Phi^0 = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \tag{34}$$

$$\Phi^* = \Phi^0 \ .* \ \Gamma^* = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \ .* \ \begin{pmatrix} 1 \\ 1.25 \\ 2.5 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 1.25 & 1.25 & 0 \\ 2.5 & 0 & 2.5 \end{pmatrix} \tag{35}$$

$$\Gamma^{max} = max_{columns}\{\Phi^*\} = \begin{pmatrix} 2.5 & 1.25 & 2.5 \end{pmatrix} \tag{36}$$

$$\Phi_{PR} = \Phi \;./\; \Gamma^{max} = \begin{pmatrix} 0 & 0 & 0 \\ 0.5 & 1 & 0 \\ 0.5 & 0 & 1 \end{pmatrix} \;./\; \begin{pmatrix} 2.5 & 1.25 & 2.5 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0.5/2.5 & 1/1.25 & 0 \\ 0.5/2.5 & 0 & 1/2.5 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0.2 & 0.8 & 0 \\ 0.2 & 0 & 0.4 \end{pmatrix} \tag{37}$$

Finally, the absolute flows:

$$Q_{PR} = \Phi_{PR} \;.*\; D^T = \begin{pmatrix} 0 & 0 & 0 \\ 0.2 & 0.8 & 0 \\ 0.2 & 0 & 0.4 \end{pmatrix} \;.*\; \begin{pmatrix} 0.5 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0.1 & 0 & 0 \\ 0.1 & 0 & 0 \end{pmatrix} \tag{38}$$

$$Q_{PR}^{in} = \sum_{in\ rows} Q_{PR} = \begin{pmatrix} 0 \\ 0.1 \\ 0.1 \end{pmatrix} \quad \leq \quad S = \begin{pmatrix} 0 \\ 0.2 \\ 0.1 \end{pmatrix} \tag{39}$$

$$Q_{PR}^{out} = \sum_{in\ columns} Q_{PR} = \begin{pmatrix} 0.2 & 0 & 0 \end{pmatrix} \quad \leq \quad D = \begin{pmatrix} 0.5 & 0 & 0 \end{pmatrix} \tag{40}$$

Maximum Throughput

$$\Phi_{MT} = \Phi \;./\; \Gamma^{*} = \begin{pmatrix} 0 & 0 & 0 \\ 0.5 & 1 & 0 \\ 0.5 & 0 & 1 \end{pmatrix} \;./\; \begin{pmatrix} 1 \\ 1.25 \\ 2.5 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0.4 & 0.8 & 0 \\ 0.2 & 0 & 0.4 \end{pmatrix} \tag{41}$$

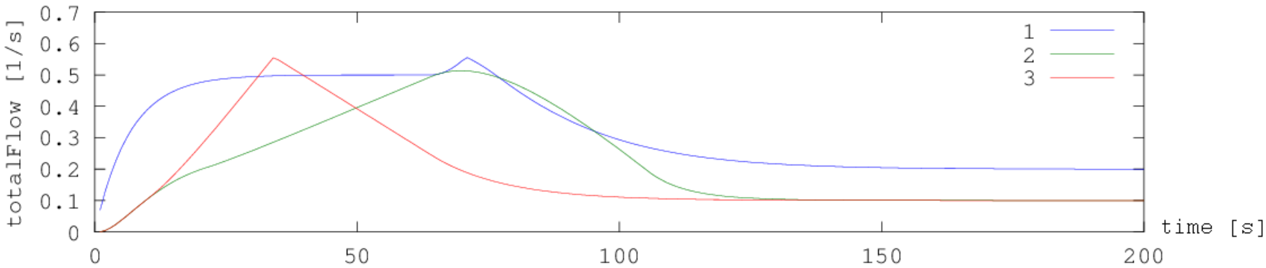
$$Q_{MT} = \Phi_{MT} \;.*\; D^T = \begin{pmatrix} 0 & 0 & 0 \\ 0.4 & 0.8 & 0 \\ 0.2 & 0 & 0.4 \end{pmatrix} \;.*\; \begin{pmatrix} 0.5 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0.2 & 0 & 0 \\ 0.1 & 0 & 0 \end{pmatrix} \tag{42}$$

$$Q_{MT}^{in} = \sum_{in\ rows} Q_{MT} = \begin{pmatrix} 0 \\ 0.2 \\ 0.1 \end{pmatrix} \quad \leq \quad S = \begin{pmatrix} 0 \\ 0.2 \\ 0.1 \end{pmatrix} \tag{43}$$

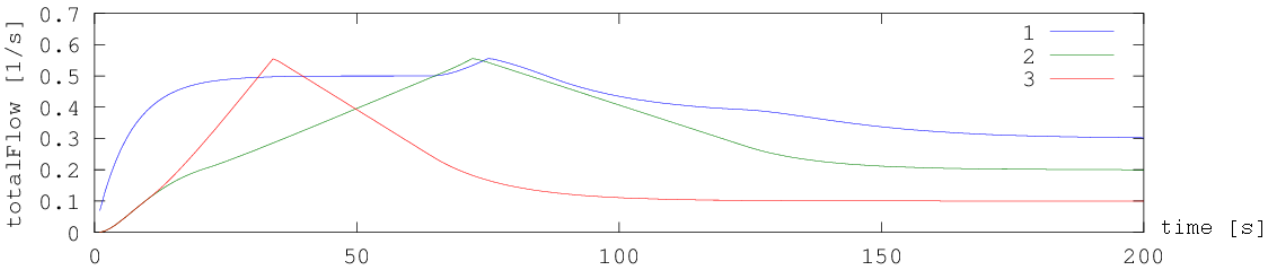
$$Q_{MT}^{out} = \sum_{in\ columns} Q_{MT} = \begin{pmatrix} 0.3 & 0 & 0 \end{pmatrix} \quad \leq \quad D = \begin{pmatrix} 0.5 & 0 & 0 \end{pmatrix} \tag{44}$$

C.4 Conclusion

The above examples demonstrate a successful application of the derived mathematical models and check the correctness of the implementation. Both algorithms satisfy the conditions of the continuity of flows (sec. 2.2). As with other submodels, the goal was to provide a generic structure allowing for use of various mathematical models of intersections, which was achieved.



(a) Proportional Reduction: the flows are reduced proportionally, the final outflows are identical.



(b) Maximum Throughput: the flows are as big as possible, the final outflows are equal to the supplies of downstream roads.

Figure 12: Simulation of the flows through an intersection. The curves present the total flows at the road sections attached directly to the intersection. After some transient stage, the inflows and outflows equilibrate and total flows stabilise on the levels predicted by the mathematical model.

D Mesoscopic model

Mesoscopic models share many of advantages of macrscopic and microsopic models. Among the listed features (see section 1.4.1) the most significant in this application is the possibility for on-line rerouteing, i.e. dynamic response to the changing conditions on the roads.

This is thanks to an autonomous character of the newly introduced entities, namely “Vehicles”. They can be characterised by:

- no interactions with other vehicles; the dynamics of a vehicle is governed by the road specific attributes, such as mean speed;
- a carried flow portion, allowing for changing its influence to the traffic conditions; by increasing the carried flow portion we can virtually represent many real-vehicles by one model-vehicle. This characteristic allows for running very efficient, lightwieghted simulations, still providing valid results;
- individual character, e.g. vehicles can be easily differentiated from each other by assigning specific attributes, among them the carried flow portions. This renders a model heterogeneity;
- finally, each vehicle can store information about the origin and destination, and therefore it can find its individual route. Most importantly, it can change at any time choosing an alternative route.

The vehicles are generated and injected into the road network at the source road sections. The number of vehicles created in each iteration depends on the value of demand specified at each source. There exist two probabilistic generators:

- *uniform* generator using the uniform probability function, which yields creation of constant number of cars at each step or, conversely, a single car once per a constant number of iterations in case of low demand;
- *Poissonian* generator utilising Poisson distribution and therefore introducing a varying number of cars into the network.

While creating a vehicle, also a corresponding portion of flow has to be produced on the road section. Each time a vehicle is transferred to a next road the equivalent flow portion have to be removed from the preceding road and inserted into the following one. Based on the total flow, the densities and resulting from them mean speeds are derived.

Each newly created vehicle calls the optimal path procedure returning the best possible route to the destination given the current conditions. The drivers' awareness of the conditions on the road may be easily determined by the form of the function returning the optimal path. Currently the Dijkstra's shortest path algorithm is performed using the travel times (defined as in section A.2 in Appendix A) as weights. This method would reflect a “perfect awareness” of drivers as they would have access to complete data (in reality, it would correspond to drivers using GPS devices showing also current congestion). There are also other possibilities such as using road lengths as weights (a GPS, no information about the congestion). Finally one can imagine also a choice based distinctly on the direction toward the destination with additional random factor (this would be a newcommer, not familiar with the city's topology). (refer to Ramming (2001) examining many possible route generation algorithms).

Similarly as in the macroscopic model (see appendix A) the Dijkstra algorithm is used for the shortest path search.

D.1 Demonstration

The following section describes a simulation demonstrating the behaviour of vehicles modifying their routes due to change of the conditions in the roads.

D.1.1 Simulation settings

A simple road network was created consisting of 5 roads as it depicted in Fig. 13. The idea was to provide two possible options, vehicles can either go through southern or northern roads. These two alternatives are of the same length, therefore a potential decision of rerouteing will be due to change of conditions, such as growing congestion. In order to enforce appearance of traffic jam the demand at the most upstream road ("east" - source) exceed the supply of the most downstream section ("west" - destination). We expect then a gradual increase of density while the roads are successively filled with cars unable to efficiently leave the zone.

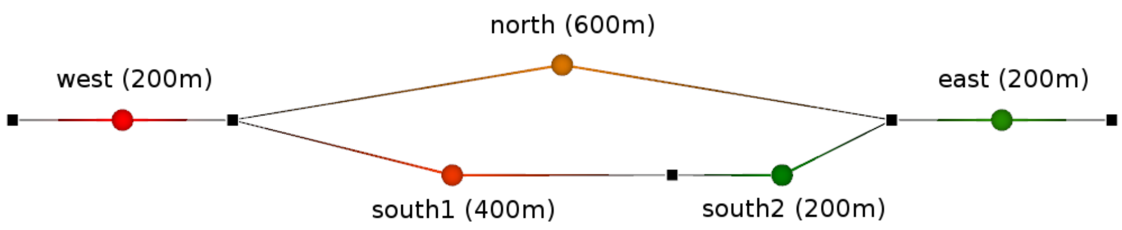


Figure 13: The road network used in the example.

In the simulations, apart from testing a proper route selection of the vehicles, two distributions were used to generate new cars: Poissonian and uniform.

D.1.2 Results

The traffic dynamics represented by densities of vehicle in the roads is presented in Fig. 14. The upper plot is related to uniform generator, which produces a certain number of vehicles every fixed number of iterations. The bottom plot corresponds to Poissonian generator. As the number of new cars varies around a mean value it was reasonable to depict results averaged over a number of simulations - the plot presented is an aggregation of 10 runs.

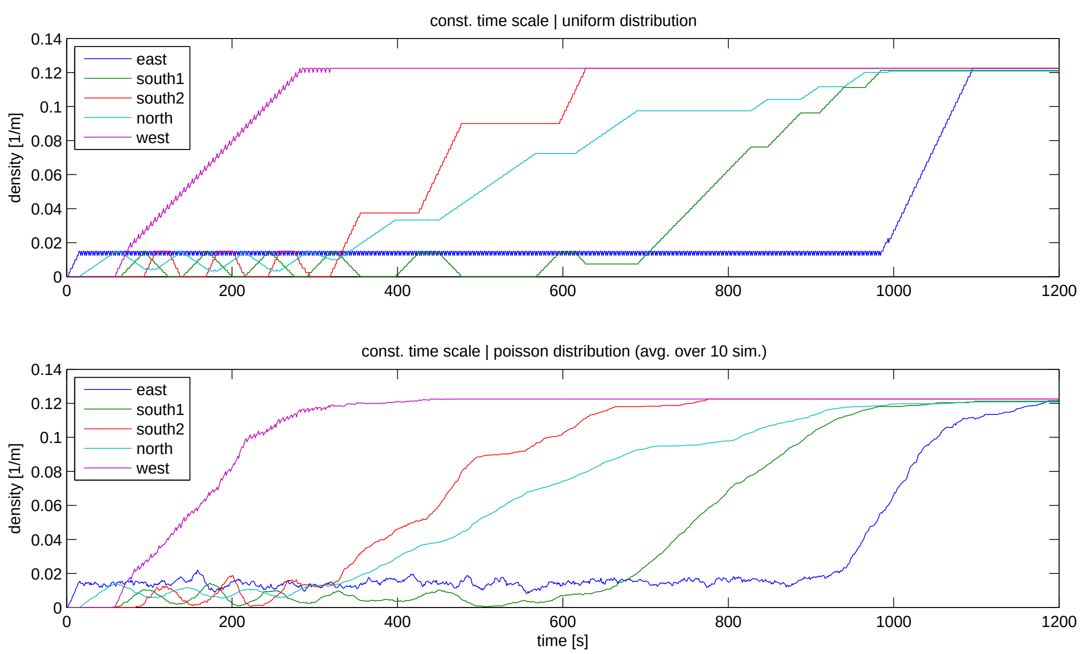


Figure 14: Comparison of the dynamics while using uniform and Poissonian vehicle generator.

Before pursuing detailed analysis it is worth to notice that for both generators the dynamics are very much alike.

In general we can observe the most downstream road (west) getting filled with car the soonest. Then a jam front expands backwards reaching and gradually filling more upstream sections. Directly after $West$ road, densities of $South2$ and $North$ branches are increasing. The growth is less intense at $North$ section. Shortly before road $South2$ gets blocked, the traffic jam reaches the preceding $South1$. Soon the densities of $North$ at $South2$ meet and approach critical level. Finally, the source road ($West$) gets congested.

Interestingly, in the plot presenting simulation with the uniform generator, one can see a sort of delayed oscillations of density in alternative roads (especially roads $North$ and $South2$ between 400-700s) while being filled. When a state of one road remains stable, the other experience the most drastic changes. In case of Poissonian generator a phenomenon is most probably smoothed out after averaging.

The slopes of the density grows differ due to roads length, as the same inflow entering a longer road would cause smaller congestion. Nevertheless, the vehicles are efficiently using both alternative routes. The north branch starts to get filled in the same moment as the south one, also they reach critical densities simultaneously.

E Variable time scales

While introducing the Cell-Transmission Model it was said it is one of the Godunov's schemes allowing for integration of the continuity equation. However, in order to converge this numerical procedure has to satisfy some conditions related to the length of time and space steps used. An equivalent of the space step is in this case the length of a road section. The condition, known as Courant-Friedrichs-Lévy (CFL) condition (Treiber, 2013), relates the time step duration between successive updates with the road's length. This section refers to the above issue in detail and describes an improved approach making use of *variable time scales.*

E.1 Introduction

Formally, the CLF condition in this case is states as follows:

$$\Delta t_{max} \max_{\rho \in [0, \rho_{max}]} (|Q'_e(\rho)|) < \Delta x \tag{45}$$

Where: Δt - time step, Δx - section's length, $|Q'_e(\rho)|$ maximum propagation speed.

The CFL condition tells in short that: *neither information can propagate over more than one cell during one time step.* If we assume that the maximum speed is given by the speed limit on a road, we can write:

$$\Delta t < \frac{\Delta x}{V_{limit}} \tag{46}$$

Therefore, given that the speed limit is the same for all roads (which is true on majority of roads in cities), the shortest time step Δt_{min} in the simulation will be determined by the shortest road section. This section has to be updated each Δt_{min}, therefore its supply and demand values would change, which, in turn, would invoke modifications in neighbouring roads and so on. Eventually entire network would need to be updated with the highest frequency.

It is clearly not an optimal way as some roads, potentially many times longer than the shortest one, could be updated less frequently.

G. Flötteröd and K. Nagel proposed in 2007 a variable time scales approach with individual, optimised time steps. A straightforward conclusion from the CLF condition is that the longer is a road the less frequently it needs to be updated. An immediate problem emerging is the communication between neighbouring cells of different time steps. A general idea is that:

- values passed from a "slower ticking" entity to a faster one are held constant (I);
- values passed from a "faster ticking" entity to a slower one are summed (II);

These rules are going to be referred to shortly while describing equations 50.

As the communication between sections is managed by connectors, they have to be invoked at least every time step ΔT_c, given by:

$$\Delta T_c = LCD(\Delta t_{S_c}, \Delta t_{D_c}) \tag{47}$$

where: LCD - largest common divisor, S_c, D_c - source and destination road section of given connector c, respectively. Δt_i is the i^{th} road's time step.

As mentioned, Δt_i should depend on the road's length, however it will prove useful to assign Δt_i with the largest power of two still fulfilling the CLF condition:

$$\forall_i \ \Delta t_i = 2^n, \ n \in \mathbb{N}_0 \text{ such us: } \Delta t_i < \frac{L_i}{V_i^{max}} \tag{48}$$

It allows to rewrite the formula for a connector's time step:

$$\Delta T_c = min(\Delta t_{S_c}, \Delta t_{D_c}) \tag{49}$$

Now instead of calculating the largest common divisor, which with arbitrary numbers rarely would allow for a significant improvement, the connectors time step T_c is given by a smaller of its adjacent cell time step durations.

Finally, one can derive the variable time scales formula for updated road's density. Analogously to equation (13) in Appendix B:

$$t = m\Delta t_i \ , \quad m = 0, 1, 2, ... \tag{50}$$

$$\rho_i(t + \Delta t_i) = \rho(t) + \frac{1}{\Delta x_i} \left(\Delta T_{s_i} \times \Sigma Q_i^{in} - \Delta T_{d_i} \times \Sigma Q_i^{out} \right) \tag{51}$$

$$\Sigma Q_i^{in} = \sum_{n=0}^{\Delta t_i / \Delta T_{s_i} - 1} Q_i^{in}(t + n\Delta T_{s_i}) \tag{52}$$

$$\Sigma Q_i^{out} = \sum_{n=0}^{\Delta t_i / \Delta T_{d_i} - 1} Q_i^{out}(t + n\Delta T_{d_i}) \tag{53}$$

where: ΔT_{s_i} and ΔT_{d_i} are time step durations of the (s)ource and (d)estinations connectors attached to road i. Otherwise notation remains consistent.

Let us explain the above formulas. First, road section i updates its state only every Δt_i iterations, otherwise stays constant. This corresponds to rule (I). The terms ΣQ_i represent buffers aggregating the meantime flows. Following the equation (49) Δt_i is always grater than or equal to ΔT_{s_i} and ΔT_{d_i}, therefore there is always at least one term under the sums. Thus, in case of road's time step duration being longer (slower entity) than the connector's one (faster entity) rule (II) is applied.

E.2 Demonstration

A simulation of the same settings as in Appendix D, section D.1 was performed. This time only the Poisson generator was used. Figure 15 presents the densities dynamics averaged over 10 simulations. The upper plot corresponds to constant time scale, while the bottom one illustrates the variable time scales.

Fig.16 shows relative deviation of number of cars when comparing VTS and const. time step results. Although there exist some peaks reaching up to 6%, in general the relative difference between densities oscillates below 1%.

Finally, the aim of VTS application was to improve the performance. A set of simulations was run with varying inflow from 0.1 to 0.6 cars per second and fixed outflow 0.2. Again Poisson generators were used and for each settings 10 simulations were executed. Fig. 17 presents an observed speedup.

For low inflow/outflow ratio vehicles do not accumulate on the road network and computation demand remains low. In this conditions simulation speed-up is negligible. However, as the ratio grows, VTS approach allows to deal with excessive number of cars far more efficiently, outperforming the const. time step method 8 times. It is not obvious why the the factor is around 8. As mentioned in section 2.3 in the main report, the longer roads accumulate also more cars, so although the longest road was updated 4 time less frequently than the shortest, the speedup could be bigger.

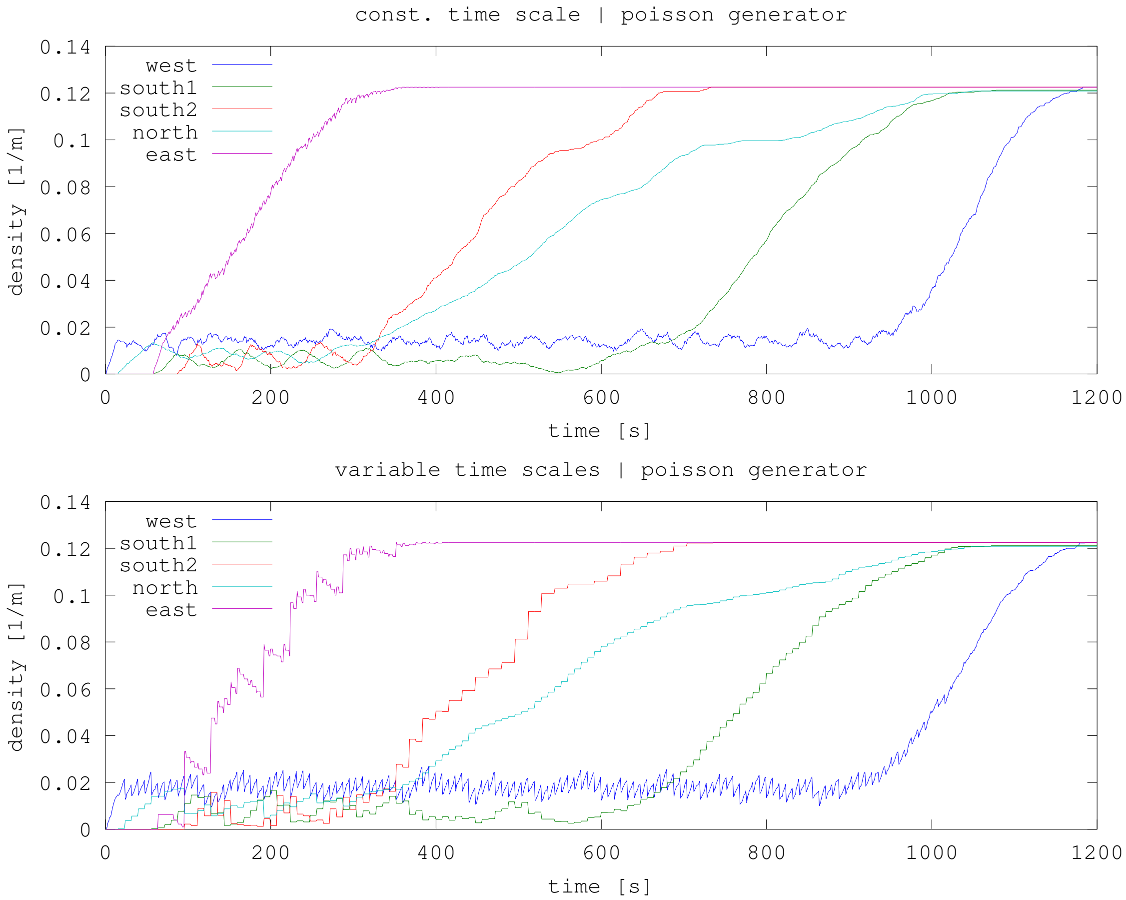


Figure 15: Comparison between the simulations using constant and variable time scales. The results are averaged over 10 simulations.

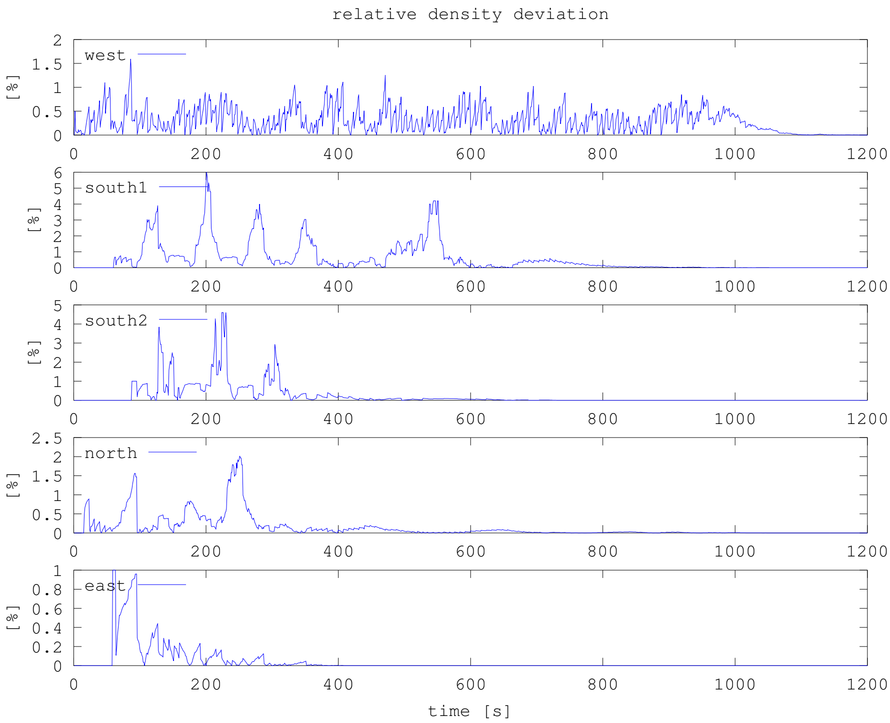


Figure 16: Relative deviation of densities when applying variable time scales with respect to constant time step simulation.

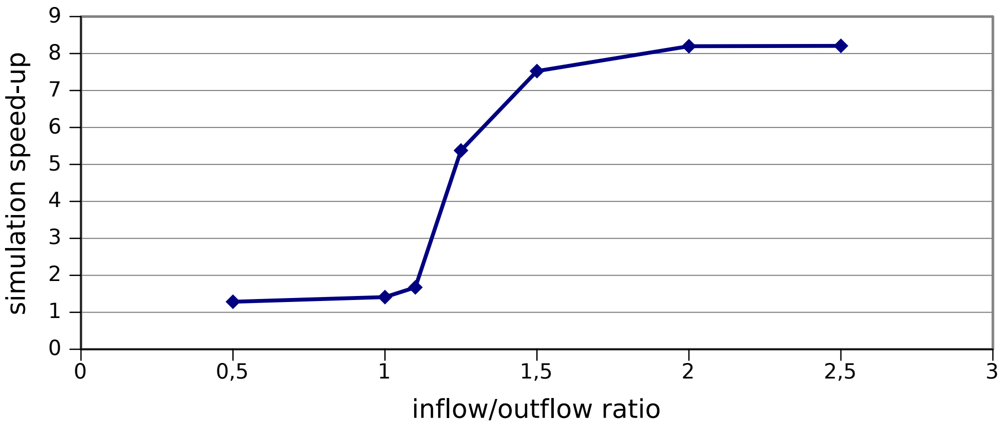


Figure 17: A speedup of the simulations thanks to Variabel Time Scales. The speedup is the biggest for high inflow to outflow ratio causing huge congestion on the roads.

F Geographic data loader

The Geographic information system (GIS) provides spatial data entry, management, retrieval, analysis, and visualisation functions[7]. Many institutions publish GIS records under open-source licences so they can be freely used. Among them, the city of Rennes provides good quality data of the road network.

The data used in the simulation was imported from the open-access webpage[8]. It consists of shape (.shp) files of the road network. Additionally there exist other files including the properties' addresses, bus lines, etc.

There are significant shortcomings of the data available:

- regardless of the real number of lines or the directions, a road is represented in the logical way, as a single curve;
- the information about the direction is available as one of the four options: 1) no-traffic, 2) direction congruent with the rise of the address numbers, 3) direction opposite to the rise of the address numbers, 4) two-way road;
- the number of lines is given not directly and only approximately, i.e. only the category of the road is know, such as for example: road of the national importance, city main road, inter-district road, local road. Given this information the most probable number of lines have to be assigned to the categories;
- the crossings of the roads are not labelled nor evidently indicated, i.e. there is no information about the intersections;

Given this, a number of alterations is needed to transform the network into the applicable format. The next sections describe the procedure in detail.

F.1 Intersections rendering

In order to explicitly extract the intersections the QGIS built-in tool is used called *Line Intersections*. It automatically creates a set of standalone points indicating intersections at every roadcross. However, the outcome cannot be directly utilise as the rendered intersections don't keep the information of the attached roads. They simply represent the geographical locations.

F.2 Primary data loader

In the previous step, apart from the existing file including road sections an additional shape file is created providing the positions of intersections.

In order to load the data a python script was written using the QGIS specific libraries (namely: qgis.core and qgis.utils). The following procedure is applied:

1. the script imports the data into temporary structures: *initialRoads* and $initialIntersections$;
2. the roads are assigned to the intersections by comparing the locations of the road endings with positions of the intersections. After being assigned the road is removed from the *initialRoads* set. It reduces the number of roads to check with each iteration;
3. afterwards, there are some roads left in the *initialRoads* set, namely the ones whose one end is not attached to any intersection - these are the terminal road sections: *sources* or *destinations*;

The previous steps provided 3 sets of entities consisting of specific attributes:

intersections : position $\{x,y\}$, set of attached logical roads;

[7] http://en.wikipedia.org/wiki/Geographic_information_system
[8] http://www.data.rennes-metropole.fr/

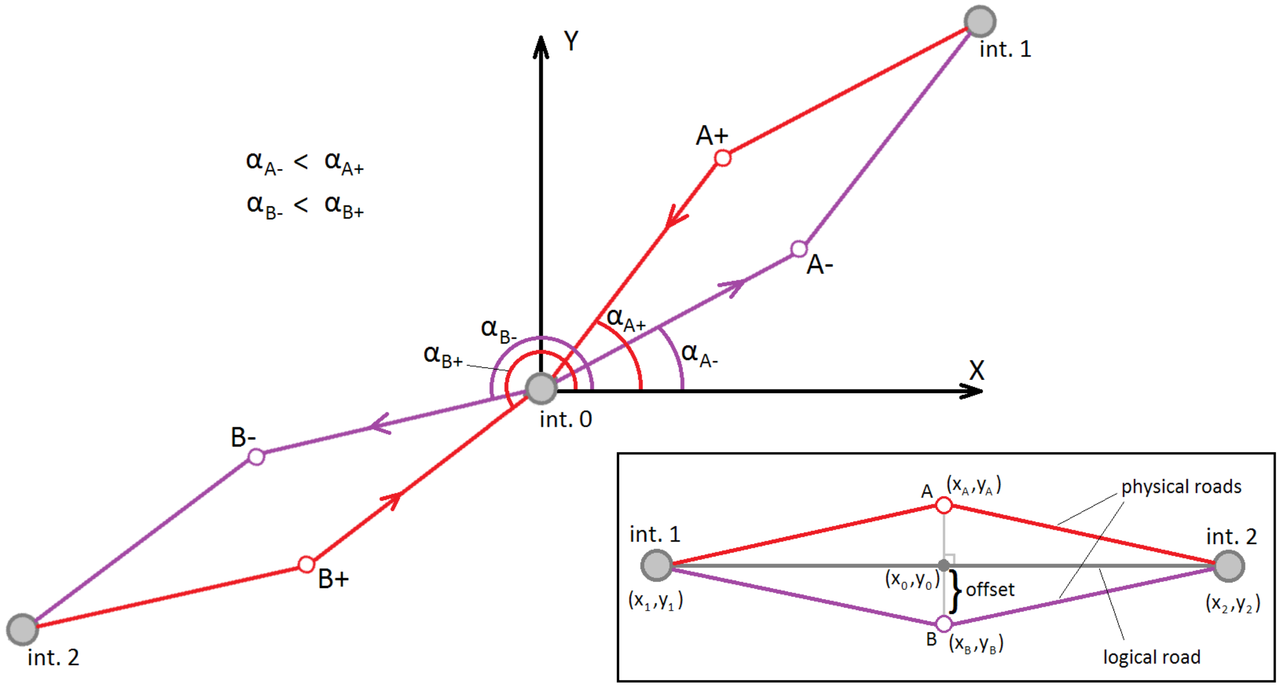


Figure 18: The physical roads

logical roads : positions of endings: $\{x_1, y_1, x_2, y_2\}$, position of the centre: $\{x_0, y_0\}$, length;

terminal roads : same attributes as above.

Next step is to obtain "physical" roads (or at least more physical, as they still do not directly correspond to the real ones) comprising also flow directions.

1. each road from the logical set is duplicated resulting in pairs of roads of the same locations. Next their centres are repositioned in opposite directions, perpendicular to the original trace as it is shown in the box in Fig. 18;
2. then, all the intersections are visited and each of the connected pairs is assessed, i.e. for each pair, the road attached at larger angle is assigned as an incoming link, while its smaller-angle counterpart as an outgoing link, see Fig. 18.

In the end a consistent directed graph with intersections as nodes and road sections as oriented links. The position alterations of the roads allows for a basic visualisation of the road network.

There is a need for one more step to complete the task of creating a road network compatible with the conceptual model. One have to create the connections between each possible pair {incoming road, outgoing road} at every intersection, so the flows of traffic between them are possible. The process follows:

1. at every intersection we loop over the incoming roads and for each of them, in turn, we loop over the outgoing roads in order to create connectors.

The outcomes of the above procedure are collections of entities with attributes:

physical roads : {length, position of the centre, source intersection, destination intersection};

sources : {destination intersection} and **destinations**: {source intersection};

intersections : {position, set of connectors};

connectors : {source road, destination road}.

Figure 19: Imported Rennes' road network

This is the minimum data needed to create a road network applicable for simulations. Based on these, the .xml files are generated (also using python scripts) that are loaded while building the instantiation model.

The figure below shows the Rennes' road network within the city's ring:

F.3 Shortcomings and further work

Although the above procedure is capable of transforming the GIS data into the CoSMo compatible format there is still significant amount of issues to be resolved before trying to perform a realistic simulation. The ones already mentioned:

- for each logical road two physical roads of opposite directions were created despite the fact that the initial ones may represent one-way streets;
- the direction of the one-way roads is defined as either "following the address numbers growth" or not. This is not a straightforward description. Although there is a database providing the addresses of the buildings along the roads a proper interpretation may be difficult in some situations, e.g. if there are no or only a single building associated with a road;
- some logical roads are represented as loops. They are common for small neighbourhoods as showed in Fig. 20. They cause problems are they virtually have only one end which results in

phantom connectors having the same road section as the source and destination, therefore not being connected to the network.

Nevertheless, the developed scripts allow for import of the data and were successfully used while testing the submodels, see appendix A section A.3.

Figure 20: Some Rennes' neighbourhoods. An example of road loops.

A complex systems platform for modeling and simulation of cities.

M. Tadeusiak*, M. Edwards, G. Barbaud, T. de Lumley

The CoSMo Company, Lyon

April 2014

Context and motivation

Due to the rapid urbanization in the last decades, more than half of the world's population is currently living in the cities. With the ever increasing complexity of the city processes linking many domains such as economy, ecology, sociology and transport, the city appears increasingly as a system being composed of many dynamically coupled heterogeneous sub-systems requiring the challenge of a revisited, more integrated modeling approach. Accurate city models prove precious to help find and decide upon optimal development choices. For a more holistic understanding of the phenomena observed in the urban areas, the next generation of models has to take into account the variation of temporal and spatial scales of the sub-systems and to provide an efficient way of presenting their coupling and their dynamics. In this context, transportation demand and resulting traffic (at different scales and linked to different fields) play a major role in the city's evolution.

The CoSMo Company is happy to offer a sophisticated tool answering the above needs in their complexity.

Methodology

The CoSMo platform is designed to best suit the complex systems specifics. The CoSMo modeling language (CoSML) enables to design a conceptual model which captures all the key aspects of the complex system being considered, allowing to explain and simulate the underlying phenomenon.

This requires indeed to define entities, which might be associated into groups, at different spatial and conceptual levels, interacting in different ways, following processes with potentially elaborate scheduling rules, leading to the emergence of global phenomena from low-level sub-models.

In CoSMo, systems are modeled as sets of interacting `entities`, each containing specific attributes and evolving following targeted information provided by their neighbors. Such entities can be dynamically created or removed by the model.

Models are not based on global control, all the interactions are local, formalized as the `environment` - a set of relations with a specific structure (i.e a graph or a regular grid) providing a means for entities to communicate. This allows to observe (bottom-up) emergence.

Hierarchical structure can be established by grouping basic entities into one or more compounds, which can be considered as higher level components. This embedding may be indefinitely repeated, providing a way to implement evolving hierarchical multi-scale systems following different axis (geographical, conceptual, ...).

The model's dynamics is managed by a `scheduler`, ordering the processes governing the evolution of the entities and the environments, as well as integrating the influence of the external factors. This scheduler can be dynamically changed by the model following the needs.

Based upon the modeling language, the CoSMo SDK platform has been created. This tool facilitates the modeling task enabling to automatically generate, from a conceptual model, a simulation

*michal.tadeusiak@thecosmocompany.com

engine that allows to run simulations on specific model instances. A simulation can be accompanied by observers keeping record of the selected measures throughout the evolution. Finally, one can use protocols managing the simulations, supporting dynamical modifications of the conditions, helping to study the model, verify its robustness and validate parameter values.

Applications

The CoSMo software was so far successfully applied to the modeling of a broad spectrum of city-related problems, e.g. the EcoCité project focusing on Grand Lyon area, performed together with Veolia and EDF. The models range from yearly scale (city planning), monthly scale (distribution network maintenance), daily scale (supply chain resiliency), down to hourly scale (distribution network crisis management, traffic model).

The Urban Planning model covers various fields: economy, population, transport and ecology. City population growth is driven by life cost and quality, itself being affected by accessibility and environmental constraints. Newcomers increase traffic and have an impact on both ambient (pollution and noise) and accessibility (traffic jams). On top of this, housing availability is driven by an economic model that is influenced by the demand for apartments (supply and demand model). The Urban Planning model operates at the scale of districts and at the time scale of years.

On the contrary, the Traffic Model regulates the flows at a time scale of minutes to hours. Depending on the granularity needed one can perform a macro simulation regarding densities and flows, or a micro simulation focusing on single vehicles. Furthermore, CoSMo permits to simulate with identical protocols (and therefore provide a meaningful comparison between) such two models of the same system but with differing mathematical descriptions corresponding to different levels of spatial precision.

Focusing on urban planning and traffic - two models related to city and transport, we can easily understand the interest to link them together. In such coupling, the Urban Planning model foresees the evolution of the city in the next decades providing the traffic supply and demand distribution. And, while in some bounds, the impact of some small traffic increase can be described by a macro model, getting just out off the bound can make slight traffic bursts over short time periods lead to major traffic jams. In such cases, a more precise traffic model can be used to get a more detailed picture, that can be re-injected into the Urban Planning model providing the information about the transport accessibility.

Conclusion

As the above example models show, CoSMo is a powerful tool to model complex systems such as the city.

Coupling multiple models helps to understand complex systems and to better predict details. A population model itself cannot anticipate the influence of traffic jams and pollution and, modeled alone, a transport model can't be used to forecast future traffic evolution without the information of population repartition.

At the same time, thanks to its modular architecture, CoSMo allows for implementation of multi-scale combined model in one simulation. While simulating one decade of traffic at hour scale is not realistic (nor useful), making a full abstraction of traffic as a transport model can lead to inaccurate results. Switching between scales on key time and spatial elements (when and where the potential traffic jams are) increases performance of a simulation making it faster or more precise where needed.

All these make the CoSMo language and platform an outstanding tool in modeling heterogeneous complex systems in a modular way and efficiently providing details of their dynamics.