



HAL
open science

On the Implementation of a Semantic Model for Intelligent Vehicle Navigation

Alessandro Corrêa Victorino, Marie-Hélène Abel

► **To cite this version:**

Alessandro Corrêa Victorino, Marie-Hélène Abel. On the Implementation of a Semantic Model for Intelligent Vehicle Navigation. 4th International Conference on Deep Learning, Artificial Intelligence and Robotics (ICDLAIR 2022), Dec 2022, Salerno, Italy. pp.47-54, 10.1007/978-3-030-85365-5_5 . hal-03523308

HAL Id: hal-03523308

<https://hal.science/hal-03523308v1>

Submitted on 12 Jan 2022

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

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

On the implementation of a semantic model for intelligent vehicle navigation

Alessandro Correa Victorino¹ and Marie-Hélène Abel¹

¹ Sorbonne universités, Université de technologie de Compiègne, CNRS, Heudiasyc UMR
7253, CS 60319; 60203 Compiègne Cedex, France
`firstname.lastname@hds.utc.fr`

Abstract. This paper considers the problem related to the navigation of multi-actuated intelligent vehicles in interaction with its occupants and its environment, by modeling and describing the semantics of the contexts associated to the movement. That is, formally defining the semantic model of the navigation context, as obstacles, pedestrians, roundabouts, information on the vehicle itself (load, power, etc.). This semantic modeling generates an additional degree of abstraction by "reasoning" about the information collected by the various sensors of an autonomous vehicle in order to build a navigation context for the latter. This navigation context is able to provide decision support for the autonomous or semi-autonomous vehicle (interaction to the driver). In this paper, we show initial modeling results by defining and selecting the relevant information in the creation of a context for the vehicle, structuring the contextual information so as to be able to reason on it and thus dynamically update the context in which the vehicle operates; maintaining information by choosing suitable data persistence models in order to be able to query previous simulations efficiently.

Keywords: Semantic navigation, context modelling, autonomous navigation.

1 Introduction

The safe navigation of an autonomous robotic vehicle performing tasks in an unknown environment requires perceiving the nearby environment, interacting with any spatial and temporal changes in this environment, and rescheduling its mission according to this dynamic. A first required functionality is the autonomous realization of movements, avoiding obstacles, in an a priori unknown environment.

Current state of works in autonomous navigation for robotic vehicles considers the displacement of the vehicle in a local context (road monitoring, obstacle avoidance, etc.) and based on the perception of the environment. In [1] a navigation method is presented by combining a visual-based navigation method with an obstacle avoidance method based on the methodology of dynamic window approach. Other works consider the global navigation problem of running a complete itinerary with the management of intersections and roundabouts [2], [3]. These navigation methodologies enable the autonomous vehicle to detect obstacles in an "empirical" manner via various sensors without however "understanding" the nature of these obstacles. Building a semantic

modeling would thus make it possible to generate an additional degree of abstraction by "reasoning" about the information collected by the various sensors of an autonomous vehicle in order to build a navigation context for the latter. Ultimately, this navigation context should be able to provide decision support for the autonomous or semi-autonomous vehicle (to inform the driver).

Context is a complex notion [4]. According to [5], it is the key element used to help intelligent entities understand how events in the surrounding world influence their behavior. It is very poorly defined, the definitions in the literature are too dependent on their own contexts [6]. The latter depend on the field of application [7], for example:

- The context is the set of circumstances surrounding an event or an object [6]. This definition is widely used in the field of psychology.
- Context is a set of explicit variables that models contextual factors in the underlying domain, for example, time, place, environment, devices, occasion, etc. [8]. This definition comes from recommendation systems.
- The context is the set of environmental states and parameters that determine the behavior of an application or in which an event occurs and is of interest to the user [9]. This definition is generally used in the field of contextual computing.

Note also that in the field of psychology, the notion of context is often used in the following sense: set of situational elements in which the object being processed is included [10].

The basic problem treated in our research is to combine these methods by taking into account a semantic model of the navigation context considering for example obstacles, pedestrians, roundabouts, crossings, information on the vehicle itself (load, power, etc.). On the one hand, it is a question of formally defining these elements of the navigation context that we can obtain, for example from the visual perception of the road and the elements around the vehicle provided by an on-board camera or through inter-vehicles communication, and on the other hand show their use in a decision support process, for example in the context of an overtaking task, emergency stop and entering and exiting roundabouts. The on-board system will thus navigate by taking into account the semantic representations of the context (itself + environment) in which it operates.

In this paper we present initial results of the semantical context of an intelligent vehicle navigation, by defining and selecting the relevant information in the creation of a context for the vehicle, structuring the contextual information so as to be able to reasoning on it and thus dynamically update the context in which the vehicle operates, and choosing suitable data persistence models in order to be able to maintain information and querying previous simulations efficiently.

2 Context definition

In order to develop a semantic context model, it is necessary to identify the elements that constitute the context associated with displacement of a vehicle.

We are interested in the spatial context. The latter can be divided into three dimensions:

- The static spatial context: information that can have an impact on the user's environment;
- The dynamic spatial context: georeferenced information collected by different types of sensors;
- The internal spatial context: information accessible from local devices such as GPS.

In the case of intelligent vehicles, it also seems relevant to integrate a temporality into the system to define periods of crowds, for example. These temporal data coupled with spatial data would allow the vehicle to predict the density of traffic that it would encounter in order to adapt its driving accordingly. This temporality of the context was not taken into account in our model but it could be taken into account, in the long term, when the driver would indicate his route on the application linked to the driving assistant.

2.1 Context modeling

The simulated navigation scenarios are generated by a python script. We decided to categorize our context in the navigation scenarios as follows, as shown in Fig. 1:

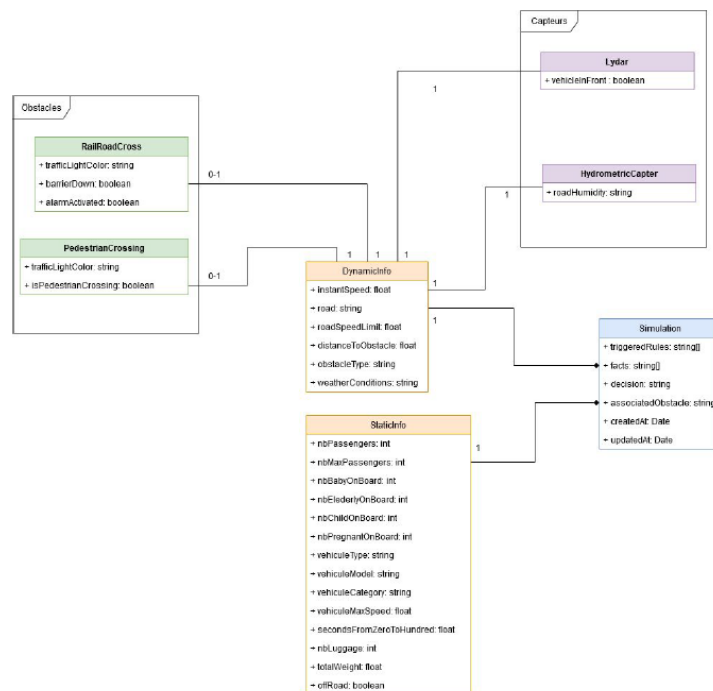


Fig. 1. Persistency of the context.

- Perennial information (StaticInfo class on the UML in Fig. 1): everything relating to the conditions in which the trip will be made. This information can be entered up-stream of the journey and may concern:

- passengers: the number, presence of elderly people, children in infants or pregnant women, etc.;
- the vehicle: type, model, technical performance, etc.;
- Real-time information retrieved from vehicle sensors:
 - the current speed of the vehicle;
 - weather conditions;
 - the type of road taken at a time t and the associated authorized speed;
- The types of obstacles that the vehicle encounters at time t :
 - pedestrian crossing;
 - Railroad crossing.

In the present case, the elements are limited to these listed above, but they could be completed.

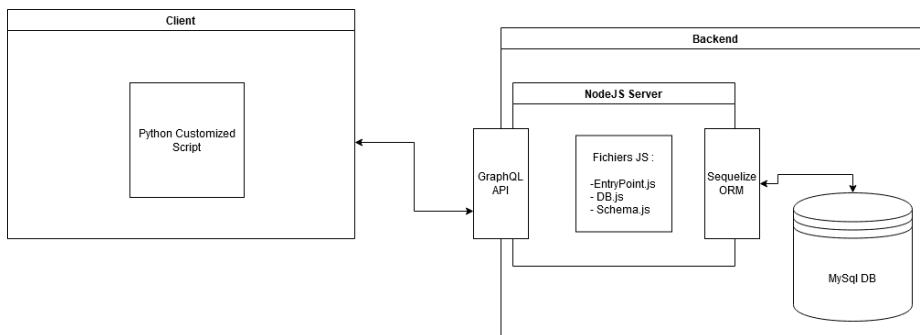


Fig. 2. Global structure of the model.

2.2 Development and technical implementation

Choice of the technologies. The technological choices to develop a version of the POC were guided by two main elements (as shown in Fig. 2):

```

1 query {
2   railRoadCrossing {
3     id
4     name
5     location
6     status
7     passengers
8     vehicleType
9     roadType
10    authorizedSpeed
11    obstacles {
12      type
13      location
14    }
15  }
16 }
  
```

```

{
  "data": {
    "railRoadCrossing": [
      {
        "id": "1",
        "name": "Pont de la Seine",
        "location": "Paris",
        "status": "active",
        "passengers": 1000,
        "vehicleType": "car",
        "roadType": "urban",
        "authorizedSpeed": 50,
        "obstacles": [
          {
            "type": "pedestrian",
            "location": "crosswalk"
          }
        ]
      }
    ]
  }
}
  
```

Fig. 3. Example of the GraphQL interface for the query to the API.

- *On the server side*, the desire to be able to efficiently query the data persisted in the database in order to easily retrieve contextual information on previous modeled simulations as well as on the vehicle's final decision-making. We therefore build the back-end on the model of an API service that could be queried from the outside to access the data. This was done using by coupling GraphQL¹ with a Node.js server given the documentation available regarding this association. Fig 3. shows the advantage of this model in retrieving data from multiple tables in a single query. Regarding the general system of data basis (GSDB), we favored a relational approach by using MySQL.
- *On the client side*, we therefore implemented the simulation generation script in python with an object-oriented programming paradigm. To tie the object-oriented programming paradigm of our script with data persistence in the MySQL GSDB, we resorted to an object-relational mapping (ORM): *Sequelize*, which is the most widely documented and preferred ORM for node.js.

Technical implementation. The development and technical implementation of the semantical navigation model is presented in a schematic way in the figure 2:

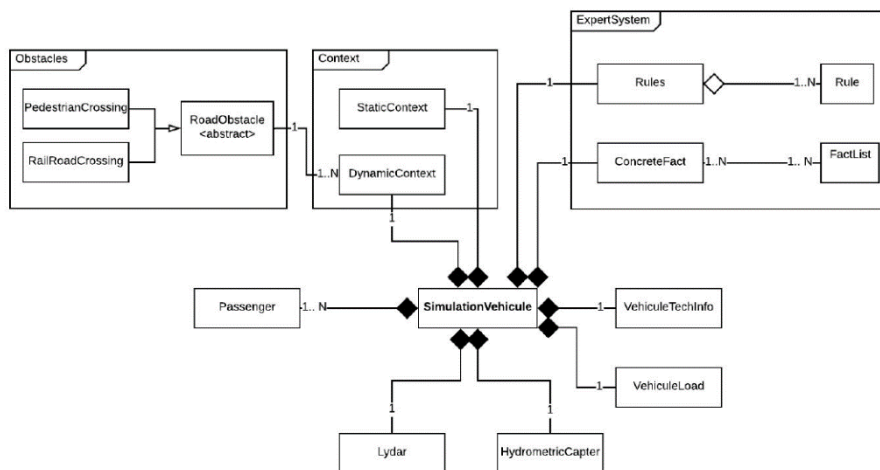


Fig. 4. UML of the global application.

Scenario simulation with Python. Each new simulation is characterized by a set of variable parameters to allow the definition of a unique context. These randomly generated contextual parameters concern, among others:

- The virtual vehicle for the simulation, taking into consideration:
 - Its type (car, motorcycle, van);

¹ Strongly typed query language developed by Facebook and under open-source license since 2015

- Its category (private vehicle, public intervention vehicle);
- Its technical characteristics (acceleration, speed).
- The obstacle encountered during the simulation
 - Each obstacle corresponds to a class with its own characteristics (presence of a traffic light, barrier, sound signal, physical retarder on the ground, delimiter, etc.)
- The number of vehicle passengers with certain specific detection criteria:
 - The age of each passenger for the elderly or frail infants;
 - Potential illness or injury;
 - Possibility for a female passenger to be pregnant.
- The load plus total weight of the vehicle:
 - Sensors associated with the vehicle during the simulation;
 - Hygrometry sensor;
 - Instant vehicle speed;
 - Lidar.

Although the data is generated randomly, it is obviously subject to verification so as not to face certain absurd situations (pregnant male passenger, number of motorcycle passengers greater than 2, etc.). We start by randomly generating a new data set for each simulation in order to build a new context for each simulation. The information generated randomly by the python client script (see Fig. 2) is specified to the user at the start of each new simulation. It is this information that we will persist in our database.

In addition to these perception data, an expert system (see Fig. 3) which will propose an action for the driver to perform according to the obtained simulation. The reasoning leading to this decision making will also be persisted.

3 Expert system implementation

An expert system is a tool capable of reproducing the cognitive mechanisms of an expert, in a particular field. It consists of three parts (see Fig. 3):

- An event basis: The information available before the intervention of the inference engine;
- A rule basis: A set of rules resulting from the expertise;
- An inference engine: That confronts the rules with the facts, allowing the expert system to conduct reasoning to dynamically deduce new facts from those already available.

The inference engine implements the following formal logic:

- Order 0: A fact is a proposition, true or false;
- 0+ order: A fact is a value;
- Order 1: A fact is a variable.

In our case, the engine is type 0 with forward chaining, that is to say, we start from the premises to draw new conclusions. These findings enrich the knowledge base and may become the premises for other rules.

A specific event is triggered based on a basis of rules.

4 Conclusions

We presented in this paper an idea in order to modelling the semantical context around a vehicle navigation. The presented semantical model uses the simulator data to identify the elements concerning the vehicle navigation at a given instant and to persist this information in order to be able to highlight the relevant information capable of identifying an obstacle. The simulation focused on obtaining different configurations of the same obstacles to generate contextual information relating to this obstacle (color of the traffic lights, positioning of a level crossing barrier, presence or not of 'a pedestrian crossing for a pedestrian crossing, etc.). An expert system capable of reproducing the cognitive mechanisms of an expert, was modelled. Based on set of events, a set of rules, the inference engine confronts the rules with the facts, in order to deduce new facts from those already available.

References

1. Danilo Alves Lima, Alessandro C Victorino. "A hybrid controller for vision-based navigation of autonomous vehicles in urban environments". *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 8, pp. 2310-2323, August 2016.
2. E. Pereira, D. Alves de Lima, A. Victorino. Autonomous vehicle global navigation approach associating sensor based control and digital maps. *IEEE International Conference on Robotics and Biomimetics (ROBIO'2014)*, Bali, Indonesia, December 2014.
3. Ide-Flore Kenmogne, Danilo Alves de Lima and Alessandro Correa Victorino. "A Global Navigation Management Architecture applied to Autonomous Robots in Urban Environments". In : *18th International IEEE Conference on Intelligent Transportation Systems (ITSC'15)*, Canary Islands, Spain, September 2015
4. Adomavicius G. & Jannach D. Preface to the special issue on context-aware recommender systems. *User Modeling and User-Adapted Interaction*, volume 24, p. 1-5: Springer.
5. Kofod-Petersen A. & Cassens J. Using activity theory to model context awareness. In *Modeling and Retrieval of Context*, p. 1–17: Springer, Berlin, Heidelberg, 2006.
6. Bazire M. & Brézillon P. Understanding context before using it. In *International and Interdisciplinary Conference on Modeling and Using Context*, p. 29-40: Springer, Berlin, Heidelberg, 2005.
7. Palmisano C., Tuzhilin A. & Gorgoglione M. Using context to improve predictive modeling of customers in personalization applications. *IEEE transactions on knowledge and data engineering*, volume 20, number 11, p. 1535–1549.
8. Adomavicius G. & Tuzhilin A. Context-Aware Recommender Systems. *Recommender Systems Handbook*, p. 217, 2011.
9. Chen G. & Kotz D A survey of context-aware mobile computing research. Technical Report. Dartmouth College, Hanover, NH, USA. 2000.

10. Bastien C. Contexte et situation. In Houdé O., Kayser D., Koenig O., Proust J. & Rastier F., Dictionnaire des Sciences Cognitives. Paris: PUF. 1998.