Most activities involved in vehicle design and development are based on CAD software, that are not just drawing and modeling tools, but also include analysis and programming features. The analysis of vehicle behaviour involves dynamic data acquisition, and a modern method consists in using GPS devices. This study propose some solutions like: an original data acquisition system based on GPS, hardware and software; a CAD application for GPS data processing, used to import and process data; applications for vehicle's dynamic behaviour estimation, using the presented devices and methods: coast-down tests, acceleration and braking tests; applications for analysis of vehicle's speed in urban traffic, using GPS devices and CAD software; an urban driving cycle proposed for Brasov city; a method, including a CAD software library, to analyse the influence of the vehicle's speed on the road traffic noise.

Dinu Covaciu
has a degree in Electronics (Politechnica University of Bucharest) and a Ph.D. in Mechanical Engineering (Transilvania University of Brasov). He has an experience of about 20 years in CAD and computer programming, and during the last years he was involved in studies regarding road traffic and vehicle dynamics.

978-3-639-35589-5

Dinu Covaciu

GPS, CAD and Vehicle Dynamics

Dinu Covaciu

Use of GPS and CAD in Vehicle Dynamics Study

A Study of the Dynamic and In-Traffic Vehicle Behaviour Using GPS data acquisition and CAD Applications

VDM
Verlag
Dr. Müller

Dinu Covaciu

Use of GPS and CAD in Vehicle Dynamics Study

A Study of the Dynamic and In-Traffic Vehicle Behaviour Using GPS data acquisition and CAD Applications

VDM Verlag Dr. Müller

Impressum/Imprint (nur für Deutschland/ only for Germany)

Bibliografische Information der Deutschen Nationalbibliothek: Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.d-nb.de abrufbar.

Coverbild: www.ingimage.com

Verlag: VDM Verlag Dr. Müller GmbH & Co. KG
Dudweiler Landstr. 99, 66123 Saarbrücken, Deutschland
Telefon +49 681 9100-698, Telefax +49 681 9100-988
Email: info@vdm-verlag.de
Zugl.: Braşov, Transilvania University of Braşov, Diss., 2010

Herstellung in Deutschland:
Schaltungsdienst Lange o.H.G., Berlin
Books on Demand GmbH, Norderstedt
Reha GmbH, Saarbrücken
Amazon Distribution GmbH, Leipzig
ISBN: 978-3-639-35589-5

Imprint (only for USA, GB)

Bibliographic information published by the Deutsche Nationalbibliothek: The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at http://dnb.d-nb.de.

Cover image: www.ingimage.com

Publisher: VDM Verlag Dr. Müller GmbH & Co. KG
Dudweiler Landstr. 99, 66123 Saarbrücken, Germany
Phone +49 681 9100-698, Fax +49 681 9100-988
Email: info@vdm-publishing.com

Printed in the U.S.A.
Printed in the U.K. by (see last page)
ISBN: 978-3-639-35589-5

TABLE OF CONTENTS

INTRODUCTION

The vehicles are used in various driving conditions, depending by the geographic area and the operating mode. It may assume that a light freight vehicle will be used mainly in urban areas, and a heavy vehicle will be used mainly on highways. Also, a car designed for urban use will have smaller overall dimensions and a low power engine, since one designed for long travels will be bigger in size and weight and with a high power engine. Basically, all vehicles should be designed according to their operating conditions. Thus a new vehicle design should start from the specifications imposed, related to vehicle destination and performances. Knowing the existent solutions with similar performances and the behaviour of existent vehicles in various conditions will be a good start. There are parameters ascertained in theoretical conditions and there are some others estimated in tables that can be found in the technical literature. Examples of such parameters are the *drag coefficient* and the *rolling resistance coefficient*.

The research activity on which this book is based consists in the following steps:

- collecting data and processing them, converting all data in a standard format;
- data analysis - this step includes the development of some custom software functions, required for analysis;
- processing and interpreting the results;
- conclusions.

The most time consuming are the steps related to data acquisition, data management and analysis.

For data acquisition were used various GPS devices, chosen according to the destination of respective data, in the same time making both the acquisition and processing steps easier. So if data are necessary for estimating the dynamic performances like braking or accelerating distances, or computing the coast-down distance, it must be used high sensitivity devices integrated in complex equipments with dedicated software. When data are necessary for traffic studies, the commercial GPS devices with a 1 Hz acquisition rate are good enough. These devices does not require special installation on the vehicle.

Data collected are saved in files of various formats, that can be stored on a server, organised following certain criteria and, furthermore, it may be attached metadata that will information about place, conditions, devices used, destination of data. Attaching of metadata can be done with client applications that communicate with the server using a file transfer protocol (like WebDAV).

Same data are loaded and processed in a CAD system with a dedicated application. This software application consists basically in a library of functions that can be called using the user interface of the CAD system. Data can be imported directly from the files saved by the GPS devices, before uploading them on a server, or can be downloaded from the server through a client application (using the same transfer protocol as above). The metadata will not affect the structure and content of the original files.

The CAD application for GPS data processing represents the core of the whole process. Using this application the following tasks are covered:

- take data from various file formats;
- transform the coordinates of the recorded points, from the geographical coordinates used by the global positioning system to rectangular coordinates, including georeferencing (corelating the ascertained data with a known reference system) of the recorded track;
- graphical representation of the track and storing the information as additional data attached to the geometric entities;
- calculation of other parameters and storing as metadata attached to geometric entities;
- graphical representation of velocities and accelerations by time and by distance;
- other 2D and 3D graphical representations;
- export of data from diagrams or tracks (represented as points and lines);
- analysis of speed versus time diagrams in order to determine a driving cycle;
- preparation of data for other applications (like traffic studies, noise mapping etc.).

The graphical representations realised inside the CAD application can be saved as CAD (*dwg*) files and can be stored on a fileserver. Data exported in various formats can be processed with external software like Excel, SPSS, MapSource or even Google Earth.

STRUCTURE OF THIS BOOK

Beside *Introduction* and *Appendices*, this book includes five main chapters:

- General aspects on studying the dynamic and in-traffic behaviour of the vehicles;
- Modern technologies used in vehicle’s performances analysis;
- Equipments and techniques for data acquisition and processing;
- Solutions in the study of vehicle behaviour;
- Conclusions and future developments.

The first chapter, **General aspects on studying the dynamic and in-traffic behaviour of the vehicles**, includes theoretical notions concerning the dynamic performances of the vehicle, the in-traffic behaviour of a single vehice and the driving cycles.

Some theoretical aspects, as the vehicle's general equation of motion, vehicle resistances, traction performances, acceleration and braking performances are considered.

Regarding the in-traffic behaviour of the vehicle, some aspects related to microscopic traffic parameters are mentioned, especially the driving speeds and accelerations, in urban cycle. These parameters will be analyses in the next chapters.

There are also presented the most known driving cycles, used as reference in Europe, USA and Japan. The standard driving cycles are useful in vehicle's performances estimation, on chassis dynamometers, but these cycles cannot describe the driving in any traffic conditions. Standardisation is useful for comparison between different vehicles, but not for establishing the real driving conditions. This leads to the need for development of custom driving cycles, based on statistical analysis.

The chapter **Modern technologies used in vehicle's performances analysis** contains a part dedicated to computer aided design and another part about global positioning using satellites.

The CAD systems offer the possibility to develop custom applications, using the application programming interfaces (API). All the important CAD systems include interfaces for C++ programming.In order to develop applications for research it may be preferred to use a more flexible programming environment, as Visual Lisp in AutoCAD. The LISP language was originally designed for artificial intelligence, but it was proven to be very appropriate for CAD applications development.

The global positioning system GPS is a system that use the signal from satellites to determine the position of the receiver on the earth (more exactly, on a reference geoid). The number of users is unlimited because the system is passive – the GPS devices are only receiving signal. The GPS receivers are able to record informations about position, velocity and time. By processing these informations it is possible to navigate using digital maps, but it is also possible to estimate some performances of the vehicle on which the receiver is installed

This chapter presents the possibilities of reading data from GPS receivers using NMEA protocol, and also the transformation of the position information from geographic coordinates (latitude, longitude) in rectangular coordinates (x, y).

The chapter Equipments and techniques for data acquisition and processing presents the equipments and methods used for collecting and processing data

The devices used are mainly GPS receivers, but also traffic classifiers, accelerometers, and some data were collected by using the OBD-II interface, reading information from the vehicle CAN Bus. The GPS receivers used are: Garmin GPSmap 60CSx, Garmin GPS 18x-5Hz and Racelogic Vbox III. It is analysed the accuracy of data recorded with these devices.

As data collecting method for road traffic studies, there are presented the method of *instrumented vehicle* and the method of *chase car* (or the mobile observer). In case of the second method, there are measured the driving parameters of other vehicles than the one with measuring equipment.

For data processing it is described an algorithm used for assessment the vehicle kinematics based on PVT data (*position-velocity-time*) obtained from GPS receivers, and the coast-down method for assessment of drag and rolling resistance coefficients.

In the chapter **Solutions in the study of vehicle behaviour** are presented the solutions identified during this research, including:

- an original data acquisition system based on GPS, which consists in a hardware part and a software program that runs on a mini-notebook which is also part of the system;
- a CAD application for GPS data processing, written in AutoLisp, runing inside the AutoCAD environment; this application is used for importing data, geometrical and graphical processing and exporting the results to other programs;
- applications for assessment of the vehicle's dynamic performances using the equipment and methods described above;
- applications for analysis the driving speed in urban traffic using GPS devices and the CAD application described above;
- proposal for a real driving cycle for the city of Braşov, Romania, based on statistics;
- use of the solutions presented above for studying the influence of driving speed on the noise generated by road traffic, and some solutions for noise mapping.

In the last chapter, **Conclusions and future developments**, the overall conclusions are listed, followed by some possible openings.

The **Appendix** contains some of the diagrams generated from the data collected and some examples of the used file formats (text, gpx).

1. GENERAL ASPECTS ON STUDYING THE DYNAMIC AND IN-TRAFFIC BEHAVIOUR OF THE VEHICLES

1.1. DYNAMIC PERFORMANCES OF AUTOMOTIVE VEHICLES

1.1.1. Vehicle’s motion resistances

The vehicle’s motion is an effect of using the energy developed by the engine, transmitted to the drive wheels. The character of the movement is determined by the size and direction of forces acting on the vehicle. A moving vehicle is subjected always to resistances caused by the interaction with the road and with the surrounding air.

When the vehicles are moving with constant speed, the tractive force is equal with the sum of all resistances. In case of accelerating vehicle, the tractive force is higher than the movement resistances, and the excess of energy developed by the vehicle engine is used to accelerate the vehicle and it is build up as kinetic energy. When braking, the tractive force is replaced by the braking force (controlled by the driver) which, together with the road and air resistances, will determine the vehicle deceleration. The kinetic energy build-up is consumed in brakes and for overrunning the road and air resistances.

The forces acting on the vehicle are [42]:

- F_R – tractive force on wheel;
- F_r – rolling resistance;
- F_p – grade resistance;
- F_a – air resistance (aerodynamic drag).

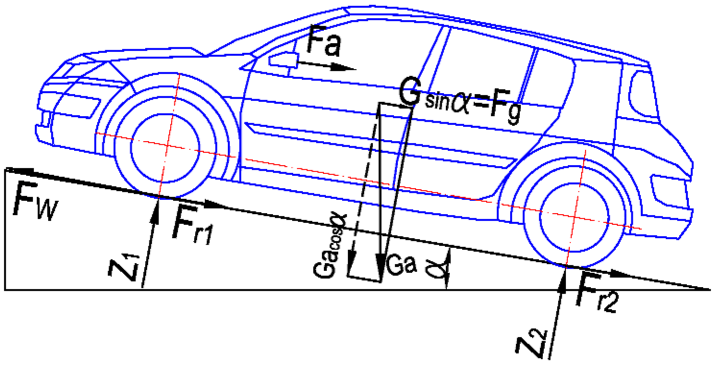


Fig. 1.1 – Schematic view of vehicle’s motion resistances

The diagram of the main forces acting on a vehicle is shown in *Fig. 1.1*.

The rolling resistance F_r and the air resistance F_a are always forces opposite to the vehicle movement. The grade resistance F_p is opposite to movement only when the vehicle is going up, and when the vehicle is going down, this force become an active force.

The complete motion resistance of a vehicle is overtaken by the tractive (wheel) force F_W and the size of this force is determined by the acceleration transmitted to the vehicle:

$$F_W - (F_r + F_p + F_a) = m_{ap} \cdot a \qquad \textit{(eq. 1.1)}$$

where: a is acceleration, and m_{ap} is the apparent mass (caused by the rotating parts, like flywheel and wheels; the vehicle acts like having a higher weight).

Rolling resistance

The rolling resistance Fr is a continuous acting force, determined by the energy loss caused by the rolling of the elastic wheel on hard or deformable surfaces [5], [53]. It is a force opposed to the vehicle's movement.

The causes of this resistance are: the tire deformation, the road deformation, the friction between tire and road, the friction inside the wheel bearings and others. The rolling resistance depends by many factors, like: tire design, travel speed, tire pressure, forces and moments acting on wheel.

The rolling resistance for a wheel depends by the normal reaction to wheel Z_i and by the rolling resistance coefficient f_i.

$$Fr = fi \cdot Zi \qquad \textit{(eq. 1.2)}$$

Considering the vehicle as a whole, the total rolling resistance is the sum of the resistances from all the wheels:

$$Fr = \sum_{i=1}^{n} fi \cdot Zi \qquad \textit{(eq. 1.3)}$$

where:

- f_i – is the rolling resistance coefficient for the wheel i;
- Z_i – normal reaction force on the wheel i;
- n – number of wheels.

On a slope, with an angle α, the rolling resistance F_r of a vehicle can be ascertained using the formula:

$$F_r = f \cdot G_a \cdot \cos\alpha \qquad \textit{(eq. 1.4)}$$

where f is a mean coefficient of rolling resistance.

2. MODERN TECHNOLOGIES USED IN VEHICLE'S PERFORMANCES ANALYSIS

2.1. USING CAD SOFTWARE FOR DATA PROCESSING

The application programming interfaces (API) of the CAD systems

The programming interfaces (API – *Application Programming Interface*) offered by the CAD systems allow users to extend, automatise and customise their functionalities. Some automation work, like generation of bill of materials (BOM), generation of drawings or parts of drawings, like some manufacturing operations, help to reduce the design time and to avoid errors. Furthermore, by integration of the intelligent systems, it is possible to improve the products quality.

All the major CAD systems available on the market offer one or more solutions for developing custom applications. In the table below are listed the most known.

Table 2.1 – Application programming interfaces available in some CAD systems

CAD System	API	Programming language
AutoCAD	Visual Lisp	AutoLisp (dialect of LISP)
	VBA (Visual Basic for Applications)	Visual Basic
	Object ARX	C++
Inventor	Open Inventor	C++
Pro/Engineer	Pro/Toolkit	C++
	J-Link	Java
CATIA V5	CATScript, VBScript	Visual Basic
	CAA-RADE	C++
Unigraphics (up to NX2)	UGOpen, UGOpen++	C, C++
I-DEAS (before NX)	Open I-deas	C++ (Orbix)
NX (after Unigraphics/ I-deas integration)	NX Open (for .NET), Open C API, Open C++ API	C (Open C); C++ (Open C++), Java, C#, Visual Basic.NET
SolidWorks	SolidWorks API	C++, Visual Basic

The most used programming language in the CAD systems APIs is C++, but many CAD suppliers are including also Java based interfaces, because of the improved portability of Java applications, or Visual Basic, because it is easy to learn.

Custom applications development in AutoCAD

Like all the major CAD systems, AutoCAD[1] offers the possibility to develop custom application, having some programming interfaces included in the standard delivery kit. So the users are able to add their own function to the existent features of the CAD system. There are three API included in AutoCAD:

- Visual Lisp, an improved variant of AutoLisp, which is a dialect of LISP;
- VBA (*Visual Basic for Application*s);
- Object ARX, consisting in libraries of C++ objects.

Each of these variants has advantages and disadvantages. The C++ language allows the development of optimized applications, with fewer resources used and with higher execution speed, but the development duration is generally longer. VBA (Visual Basic with specific AutoCAD functions included) has the advantage of the ease of learning for the programmers who are already familiar with implementations of Visual Basic language for other applications. Visual Lisp allows quick development of applications and quick query of geometric objects, directly from the command prompt. The compatibility *AutoLisp/Visual Lisp* between different AutoCAD versions is almost complete.

Object ARX

The ObjectARX programming environment offer an object oriented programming interface (it is based on C++ language) that allow programmers to customize and to extend the AutoCAD features. The ObjectARX libraries contain a complex set of tools that enable access directly to the AutoCAD drawing database and to the functions that control the graphic system. It is also possible to define new commands.

Furthermore, the ObjectARX libraries can work together with Visual Lisp, ActiveX and COM functions, so that the programmers can choose the development tools that are more appropriate for their needs and experience.

ObjectARX is useful for:

- access to AutoCAD drawing database;
- communication with AutoCAD editor;
- development of user interfaces using MFC (*Microsoft Foundation Classes*);

---

[1] AutoCAD is registered trademark of Autodesk Inc.

- development of custom classes (C++);
- development of complex applications;
- interfacing with other programming environment.

AutoLisp / Visual Lisp

Many years, AutoLisp was the standard method for developing custom applications in AutoCAD. AutoLisp is based on LISP programming language, created at the end of years ’50. LISP is a programming language designed initially for artificial intelligence and it is still used for that kind of application. AutoCAD introduced AutoLisp as API in his release 2.1, at the middle of ’80. It was chosen the LISP language as the first API variant because it was the best method for processing non-structural data inside the CAD projects [10].

The main feature of LISP is that it is list oriented (the name of the language come from *LISt Processing*). The lists are enclosed in parantheses, and the parantheses can be combined in an imbricate structure.

Example:

```
(list1 (sublist1 (sub-sublist1) (sub-sublist2 (....))...)
(sublist2 ...))
```

A list can be composed by any number of elements, but it can consider that any list has two parts: the first element and the rest of list. Any of these two parts can be a basic element (also named *atom*) or another list, which also can be composed by two parts that can be two lists and so forth.

The two parts of the list are called using two basic LISP functions: *car* and *cdr*. The names of these functions are originated from the names of two registries of the first computer on which was implemented LISP. *Car* calls the first element of the list and *cdr* calls the rest of the list.

Another important feature of LISP is that the functions, no mater if there are basic functions of the language or custom functions defined by the programmer, have the same list structure. The first element of the list is the function name and the rest of list consists in the parameters sent to the function. If the function has no parameters, the second part of the list (*cdr*) will be void, or in LISP language *nil*.

The LISP operators behave similar to functions - the operator is the first element in a list and the rest of the list represents the parameters.

Examples of using the operators:

```
Command: (+ 2 3)
5
Command: (* 5 6 23 1)
690
```

```
Command: (/ 2 3)
0
Command: (/ 2.0 3)
0.666667
Command: (cdr '(+ 2 3))
(2 3)
Command: (car '(+ 2 3))
+
```

Examples of using the functions:

```
Command: (setq lista (list 2 3 "a"))
(2 3 "a")
Command: (nth 1 lista)
3
Command: (setq lista (list 1 lista))
(1 (2 3 "a"))
Command: (append lista (list lista))
(1 (2 3 "a") (1 (2 3 "a")))
```

An advantage of programming in AutoLisp, as can be seen in the examples above, is that the LISP expressions can be written in AutoCAD directly at the *Command prompt*.

The new generation of LISP language implemention in AutoCAD is called Visual Lisp (or VLISP), which add more features to the old AutoLisp language. VLISP extends the language with an interface for *Microsoft ActiveX* objects and improves the possibilities of AutoLisp to respond to events by implementing special functions called *reactors*. As development environment, VLISP comes with a full editor and a compiler (the compilet code is executed only inside AutoCAD), a *debugger* and other useful tools for increasing the productivity of custom applications development for AutoCAD.

Visual Lisp has his own set of windows and menus, but does not run independent by AutoCAD. In order to run applications from VLISP environment it is necessary to interact with the grafic interface and with the command line of AutoCAD.

All the existing functions in previous AutoLisp versions are maintained also in Visual Lisp, so that the old applications written for AutoCAD R14, R12, R10 or even older, are fully compatible with AutoCAD 2000 - 2011.

Accessing the geometric entities through AutoLisp / Visual Lisp

As any high level programming language, AutoLisp (Visual Lisp) includes functions for controlling variables (assignment, type conversions), arithmetic and boolean functions, test and cycle functions. It is not necessary to specify the variable type from the beginning, the memory allocation being dynamically, according to the value assigned to the variable. AutoLisp is in *interpreted* language (compiled into machine readable code at the moment it is run by the language interpreter). The programs run slower than the compiled programs but the developing time is shorter.

AutoLisp (Visual Lisp) allows the access to the geometric entities stored in the AutoCAD drawing database using lists [10]. These lists have a different structure depending by the entity type. As example, for a simple line, the structure of the list may be as follows:

```
 ((-1 . <Entity name: 7ecb3dc0>) (0 . "LINE") (5 . "5916D8") (102 .
"{ACAD_XDICTIONARY") (360 . <Entity name: 7ecb3dc8>) (102 . "}")
(330 . <Entity name: 7ef78cf8>) (100 . "AcDbEntity") (67 . 0) (410
. "Model") (8 . "0") (62 . 1) (100 . "AcDbLine") (10 -32088.9
170213.0 0.0) (11 -32356.2 170327.0 0.0) (210 0.0 0.0 1.0))
```

where the first element in each sublist is a code identifying the property and the rest of the list contains the value(s) corresponding to that property (entity type, color, layer, start point, end point, etc.).

The access to AutoCAD entities data is made using the function *entget* for reading, and for creating or modifying data can be used the functions *entmake, entmod, entupd*.

The access and control of the entity data in AutoCAD is possible by following the steps described below:

- select an AutoCAD entity (geometric object - line, polyline, text etc.) and assign a name to that entity

  ```
  (setq ent (car (entsel)))
  ```

  ent is the name of a variable, *car* is the function used to read the first element in a list, *entsel* is the function used to select a single entity on the screen; the result is a list containing the entity name and the point (the *x,y,z* coordinates) used to make the selection; *setq* is the assigning function;
- read the list of entty data

  ```
  (setq el (entget ent))
  ```

  el is the name of the data list, *entget* is the function used to read the data of the entity given as parameter;
- extract the value of desired property from the data list; it can be considered that the list *el* is composed by sublist of type *(name value)*, one sublist for each property of the selected entity.

  ```
  (setq tip (cdr (assoc '0 el)))
  ```

  tip is the name given to the variable used to store the value – in this example it is the type of the geometric entity (line, point, text etc.); *cdr* is the standard AutoLisp function for reading the rest of the list, after extracting the first element; `'0` indicate the sublist with „0” as first element. To read the coordinates of the end points of a line, it can be used the sequence:

```
(setq p1 (cdr (assoc '10 el))
      p2 (cdr (assoc '11 el)))
```

- to modify a property, it can be used the function *subst* for replacing one of the sublists, then *entmod* to refresh the current drawing database.

A drawing in AutoCAD is the working DWG file, containing the database with the standard properties of AutoCAD objects, and also the user defined properties.

When the objective is to access data and/or change more entities through a single function, it may be used the auxiliary command *select*. This command is called inside various standard edit commands, like: *copy, move, erase, chprop*. A common characteristic of these edit commands is the prompt *Select objects*. It is expected to select more geometric entities. The relevant AutoLisp function is *ssget*.

The result of *ssget* function calling is the creation of a selection set. A selection set can be managed using a specific set of functions, different by the list management functions.

For creating a new selection set or for adding a new item into an existing selection set it is used the function *ssadd*. For extracting an element from the selection set, the function is: *ssname*.

Example:

```
Command: (setq ss (ssadd))
<Selection set: 16de>
Command: (setq ss (ssadd (entlast) ss))
<Selection set: 16de>
Command: (setq ent (ssname ss 0))
<Entity name: 7d7f8888>
```

Advanced data management functions

Beside the standard properties, the user can attach custom data sets to the AutoCAD entities. There are two way for attaching auxiliary data: **XData** (*extended entity data*) and **ActiveX** – through accessing data as ActiveX objects.

XData [9] is the older method, introduced by older AutoLisp versions, starting with AutoCAD Release 11 (before Visual Lisp). This method implies the use of extended DXF codes. Data are organised in lists with a DXF code as first element (identifier). When querying the AutoCAD entity with *entget*, these lists appear after the regular data list.

Examples of extended DXF codes and their description are given in *Table 2.2*.

The extended data lists can be extracted using *entget*, like the standard data. The beginning of extended data is marked with the code *-3*, preceding the first group *1001*. This group contains the application name attached to the selected entity. There can be more applications attached to the same entity and consequently can be more codes *1001* in a single list of extended data.

In order to use an application, the application name must be registered as extended data for each entity and also in a special table named *APPID*. The registration can be done using a special function named *regapp*, having the name of application as parameter. If the returned result is *nil*, this means there is already an application registered with that name. If the returned result is the name of the function, the application was registered successfully.

Table 2.2 – Extended DXF Codes

((-1, -2	entity name)	standard definition of an entity
(0 ... 239	regular data fields)	
)		
(-3	extended data	extended data attached to entity
(1001	name of application 1)	
(1000,		
1002 ... 1071	extended data fields)	
....		
(1001	name of application 2)	
(1000,		
1002 ... 1071	extended data fields)	
....		
(1001	name of application 3)	
....		
)		

The extended data attached to an entity, related to a registered application, can be read as follows:

```
(entget (car (entsel)) '((„nume_app”)))
```

where *(car (entsel))* is used to select one entity in the drawing and *nume_app* is the name of the application that controls the extended data. The size of the extended data list is limited to 16 kilobytes for each entity. This space can be managed using two functions: : *xdroom* and *xdsize*.

Another method for attaching additional data to the drawing entities uses **ActiveX** objects and functions.

The ActiveX programming interface is used in various programming languages and environments. When working with ActiveX objects in Visual Lisp, the same object models, properties and methods that can be accessed also from other programming environments are used. The objects are the basic elements of an ActiveX application. The geometric elements, like lines, arcs, circles, polylines, points or text, can be referred as ActiveX objects. In addition, other AutoCAD components are represented as ActiveX objects:

- style settings, like the line types and dimension styles;
- management structures, like layers, groups and blocks;
- the drawing window (the viewport);
- the 3D modelling space and the paper space (*model-space*, *paper-space*).

Even the AutoCAD application and drawing are treated as objects.

Comparing to the AutoLisp standard functions used to access the entities, the ActiveX functions runs faster and allow an easier access to the object properties.

The ActiveX functions have the prefix *vl-*, *vla-*, *vlax-* or *vlr-*. The functions in the *vlr-* group work with a special category of objects, called *reactors*, and can be used to define the reaction of the software to some specific events (like changing in position or dimension of a line or another geometric object).

As example, the radius of a circle can be read in the classic way:

```
(setq radius (cdr (assoc 40 (entget circle-entity))))
```

Using the ActiveX functions, the same operation can be written as follows:

```
(setq radius (vla-get-radius circle-object))
```

The function used for attaching custom data to entities is *vlax-ldata-put* and can be called as follows:

```
(vlax-ldata-put e key data)
```

where *e* is a variable identifying the entity, *key* is the property name and *data* is the actual data (the value) – so data are registered as <*name, value*> pairs.

The additional data attached to an entity can be as complex as needed and must be organized as lists. The size of the data list attached to an entity is automatically limited by the available memory. A drawing containing many geometric entities with complex additional data attached will need more time to regenerate, sometimes leading even to a system crash.

Data can be read using the function *vlax-ldata-get*, like in the example below:

```
(vlax-ldata-get e key)
```

and the returned result is the value associated with property name *key*, for the entity *e*.

This method was preferred by author for storing and processing in AutoCAD the data recorded with various GPS devices installed on different vehicles.

2.2. GLOBAL POSITIONING SYSTEM

The Global Navigation Satellite Systems (GNSS) allow an accurate determination of the position in a geocentric reference system, anywhere on the Earth, using the signal from the artificial satellites network Navstar-GPS (USA), Glonass (Russia) and Galileo (Europe).

2.2.1. GPS - Overview

GPS (*Global Positioning System*) was originally developed as a military positioning system, but it has significant benefits for civilian use as well. After some years of applying an intentional degradation of the signal by US Department of Defense, for civil users (selective availability), the restriction was suspended and the accuracy is now very good. The number of users is theoretically unlimited, the system being passive – the user devices are only receivers.

GPS consists in three major segments: the space segment, the control segment and the user segment.

The space segment consists in the satellites transmitting information about their position. In order to cover any position on Earth there are at least 24 satellites needed. In this condition, 4 to 10 satellites should be visible from any point, at any time. The current number of satellites is more than 24 – the system was officially declared as full functional in 1995, and in 2001 were 29 satellites in use [22]. At the time of writing this study there are 32 GPS satellites launched, and 27 from these are in operation. These satellites are placed on six circular orbit planes, with an approximative 55 degrees inclination, at an altitude of 20,200 km. They have a rotating period of about 12 hours and a speed of about 3,9 km/s. The satellites have an average life duration of 7,5 years (the new generation satellites have an operating duration of up to 15 years [22]) and are equipped with atomic clocks and two radio transmitters in D band. The two carrier frequencies are f1=1575,42 MHz and f2=1227,6 MHz. The signals are emitted in the spread-spectrum technique and use two pseudo random codes: one C/A (*Coarse Acquisition*) code on f1 frequency and one P (*Precise*) code on the frequencies f1 and f2 [20].

The ground control segment transmits the location parameters, controls the paths and the data transmitted and changes the satellite orbits. This segment consists in a master station (located in Colorado, USA), an alternate control master station and a network of monitor stations and dedicated antennas, spread on the globe. The positions of these stations are known with a very high accuracy. Data collected from the monitor stations are transmitted to the master station for processing. The results include the satellite positions as functions of time, the almanac, atmospheric data, satellites clock parameters and others. These data are sent to satellites using the control stations.

The user segment consists in all the GPS devices that receive simultaneously the signal from the visible satellites and calculates the solution of the navigation equation, PVT (Position, Velocity, Time). The signals from at least four satellites are needed to obtain the solution. Based on the satellites navigation parameters it is possible to calculate the distances R_i (*Fig. 2.1*) between each satellite i and the GPS receiver [20].

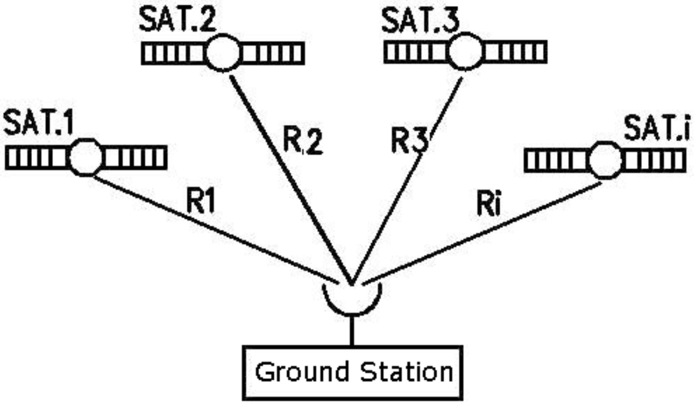


Fig. 2.1 – Simplified diagram of the GPS system

Using a GPS receiver installed on a rigid body (like an automobile) it is possible to ascertain his position on Earth. Having two or three receivers (or a dedicated receiver with two or three antennas) on the same rigid body, it is possible to ascertain the orientation, on a plan or in 3D space.

The determination of position using GPS can be affected by some errors, like [15], [47], [72]: orbital errors (also known as ephemeris errors, these are inaccuracies of the satellite's reported location), ionosphere and troposphere delays, signal multipath (occuring when the GPS signal is reflected off objects such as tall buildings or large rock surfaces), receiver clock errors (a receiver's built-in clock is not as accurate as the atomic clocks onboard the GPS satellites), number of satellites, satellite geometry/shading (this refers to the relative position of the satellites at any given time).

Because of these errors, the absolute coordinates on the ground are calculated with an accuracy given in meters. However, it is possible to obtain high accuracy (centimetres or milimeters) in relative coordinates (using two receivers) or by receiving a correction (differential) signal from a fixed station (DGPS).

GLONASS

GLONASS (*GLObal NAvigation Satellite System*, or in russian: *GLObalnaia Navigationnaia Sputnikovaia Sistema*) is a satellite positioning system developed initially by Soviet Union and continued by Russia [81]. It was designed as an alternative to the US GPS system (and to the European system Galileo, still in development).

The GLONASS design was started in 1976, and between 1982 and 1995 the satellites were launched to the orbit. Between 1995 and 2001 the system was not maintained because of the economic difficulties, and in 2001 was initiated the system repair. From 2007 India becames partner in the program.

The complete system will include 24 satellites at an altitude of 19100 km, grouped in 3 orbital planes with an angle of 120°. From any place on Earth should be visible 5 satellites.

The accuracy given by the full functional system is expected to be of 57-70 meters horizontally, 70 meters vertically, and about 15 cm/s in velocity error.

In March 2008 the system was not functional but it was active, and 16 satellites were operational. In February 2011 there are 26 satellites in constellation, 22 of them being operational and 4 in maintenance.

The Galileo System

Galileo is the satellite positioning system developed by the European Space Agency [80], as alternative to GPS. This will assure the Europe autonomy in satellite positioning, since GPS is controlled by US military authorities which may introduce again the intentionally degradation of signal, any time. In the same time, Galileo will assure full compatibility with GPS. The system will be functional probably in 2015. Together with the EGNOS[1] service it is estimated a positioning accuracy offered by Galileo of about 2 meters.

The Galileo system will use 30 satellites (27 operational and 3 spare satellites) rotating on three circular orbital plans, with an inclination of 56°, at an altitude of 23,222 km. They represent the space segment.

The ground control segment is composed by two control centers and a network of stationary radio stations. The ground stations monitor the quality of the signal received from satellites.

---

[1] EGNOS – European Geostationary Navigation Overlay Service – was officially launched at October 1st 2009. The system consists in three geostationary satellites (above Atlantic Ocean and Europe), about 40 ground radio stations and 4 control centers. These improve the quality of the signal received from the GPS satellites, and in future will work in the same way for Galileo. The user access to the EGNOS signal is free.

The user segment consist in the mobile receivers, similar with the GPS receivers already in use. The GPS and Galileo receivers should be fully compatible.

It is expected an improved quality of positioning with Galileo, with an accuracy of 4 meters without differential signal (a correction signal sent by the ground radio stations) and 10 cm when the differential signal is available.

Since at the time of elaboration of this study the GLONASS and Galileo systems are not full functional for users, only GPS devices were used.

2.2.2. GPS Signal Reception

The block diagram of a GPS receiver [17] is shown in *Fig. 2.2*.

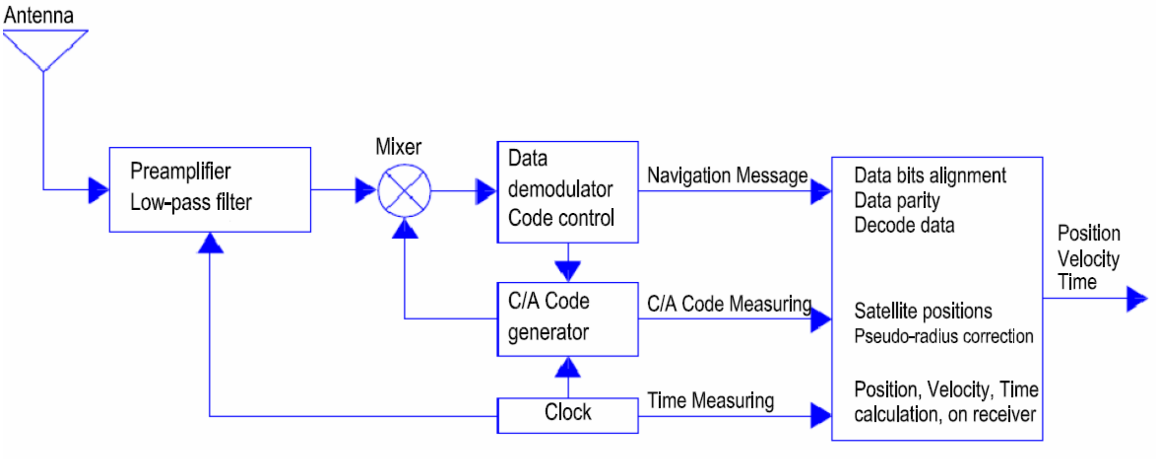


Fig. 2.2 – Simplified block diagram of a GPS receiver

Basically, the signal received by antenna is taken by a preamplifier with a low-pass filter, to reduce the higher frequency noise and to amplify the useful signal, with a carrier frequency over 1 GHz (section 2.2.1). The signal is then mixed with the *C/A* code generated locally and introduced in a demodulator for extracting the navigation data.

The GPS satellites transmit signals on two carrier frequencies, offering two levels of service: SPS and PPS.

The SPS (*Standard Positioning Service*) signal is received by most of the commercial devices. The quality of the SPS signal may be altered intentionally by the US *DoD* (*Department of Defense*). The SPS signal is transmitted on the f_1 frequency, which is 1575.42 MHz.

The PPS (*Precise Positioning Service*) signal offers a higher accuracy and it was initially resered for military and special applications. The signal is transmitted on the f_2 frequency, which is 1227.60 MHz, and it is used for ionosphere and troposphere delays measurement.

The two carrier frequencies are phase modulated using three binary code signals:

- The *C/A* (*Coarse Acquisition*) code modulates the phase of the f_1 carrier;

- The *P* (*Precise*) code modulates both carriers, f_1 and f_2;

The navigation message modulates also the f_1 carrier signal. The navigation message is a 50 Hz signal consisting in data bits describing the GPS satellite orbits, the time corrections and other system parameters.

In *Fig. 2.3* [17] is can be seen how these signals are mixed and modulated.

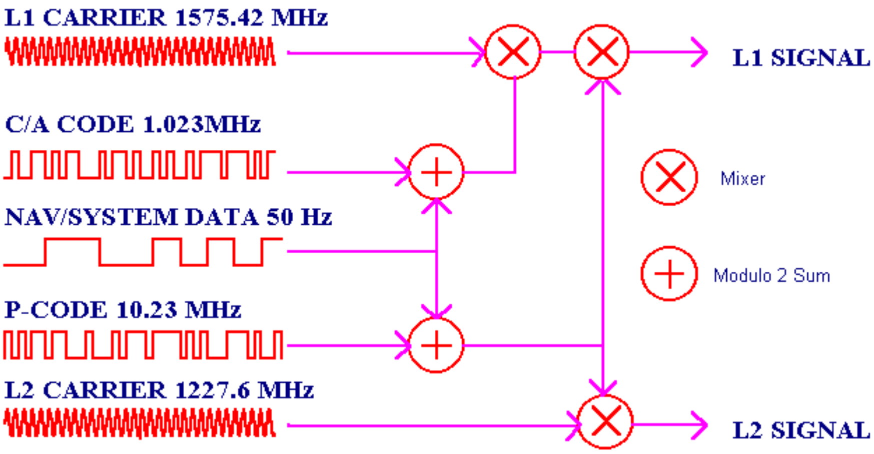


Fig. 2.3 – Signal modulation before transmision [17]

At reception, based on the three signals, the PVT (*Position, Velocity, Time*) information is calculated (*Fig. 2.2*) and sent further to the navigation device, using NMEA or similar protocols.

2.2.3. Reading Data from GPS Receivers. NMEA Sequences

The GPS receivers can store data in their own non-volatile memory (like *SD* cards or *CompactFlash*), in text files using a *txt* or *gpx* format, or they can send data to other devices using RS232 or USB interfaces and a certain transfer protocol.

The most known transfer protocol used for transmitting GPS data is the standard recommended by NMEA (*National Marine Electronics Associations*). The NMEA 0183 [66], [79] standard defines the requirements for the electric signal and for the data transmission protocol, and also the specific instructions format for a serial data bus of 4800 baud rate. For higher transmission rates (up to 38.4 kBd) it was defined an extension of the protocol, NMEA 0183-HS (*High Speed*). The NMEA standard is not intended only for communication with the GPS devices, but also for other electronic devices used in marine applications.

The informations sent on a NMEA 0183 protocol basis will be named further *NMEA sequences*.

A NMEA sequence consists in a character string composed by a type identifier and more data fields, comma separated. The type identifier is used at reception for establishing the type and format of the data sequence that is to be read. The number and length of data fields depends by the sequence type.

For communication with GPS devices, the NMEA sequence identifiers start with „GP" characters. Some of the NMEA sequence identifiers are given in *Table 2.3*.

Table 2.3 – NMEA sequence identifiers

Identifier	Description
GPALM	Almanac Data
GPDTM	Used Datum (eg. WGS84)
GPGLL	Longitude/Latitude
GPGGA	Data used for 3D positioning and accuracy
GPGSA	General satellite data
GPGSV	Detailed satellite data
GPRMC	Minimum recommended GPS data
GPVTG	Heading and ground speed

The main NMEA sequences used in the application described in this book are presented in the next pages with examples. All the identifiers start with the symbol $.

GPGGA contains essential data for positioning and accuracy information. Example of a GPGGA sequence:

```
$GPGGA,123519,4807.038,N,01131.000,E,1,08,0.9,545.4,M,46.9,
M,,*47
```

The meaning of the fields in the above sequence is:

$GPGGA - Identifier: 3D position data
123519 - Time when the record was taken: 12:35:19 UTC
4807.038,N - Latitude: 48 degrees, 07.038 minutes, North
01131.000,E - Longitude: 11 degrees, 31.0 minutes, East
1 - Quality of positioning
  - 0 = not valid
  - 1 = GPS precision (based on SPS signal)
  - 2 = DGPS precision (differential signal)
  - 3 = PPS precision (PPS signal – military and special applications)
  - 4 = RTK (Real Time Kinematic)
  - 5 = RTK floating
  - 6 = estimation

3. EQUIPMENTS AND TECHNIQUES FOR DATA ACQUISITION AND PROCESSING

3.1. DEVICES USED FOR DATA ACQUISITION

In the last years, the GPS technology became common and popular. On the market are offered various applications, especially for navigation and for recording of the route travelled. Combining position information obtained from GPS with detailed digital maps, it can find the desired destinations and the optimum routes to follow.

The decrease in price of electronic-devices and the precision offered by the GPS, even for commercial applications, encourage the apparition of more and more new applications.

For the vehicle dynamics study can be used GPS devices, installed on vehicles. In this chapter are presented the results of tests made during the last three years, in order to verify the GPS devices precision and also for research purposes.

3.1.1. GPS devices

On experiments, there were used three types of GPS receivers, able to record data during the travel: Racelogic VBox, Garmin GPSmap 60CSx and Garmin GPS 18x-5Hz. The technical characteristics of these devices are presented on the further pages. For validation, data obtained from the GPS devices were compared with data collected with other devices.

Racelogic Vbox III (*Fig. 3.1*) is a professional device with a recording rate of up to 100 samples per second, able to compute speed and acceleration and to graphical represent the gathered data in real-time [49]. It is designed especially for measuring the position and velocity of a moving vehicle.

Beside the regular data supplied by a GPS receiver, VBox is able to calculate and record (in a proprietary format output file with *.vbo* extension) the travelled distance, acceleration and other parameters. The *vbo* file format is presented in appendix 6.2.1.

Thanks to the possibilities of connecting with various external modules, other data can be also acquired; as example, Vbox can be connected to the vehicle CAN Bus, or can be connected with accelerometers.

Vbox III is very easy to install and records the collected information on a

3.2. METHODS FOR COLLECTING AND PROCESSING DATA

To measure the dynamic performance of a vehicle, no matter what instruments or tools are used, it is required to install these instruments on the vehicle and to collect data on the move.

The traffic data can be collected either by using instruments installed on moving vehicles (*instrumented vehicle* or *chase vehicle*), either by using stationary equipment, installed on the roadside (radar, video camera).

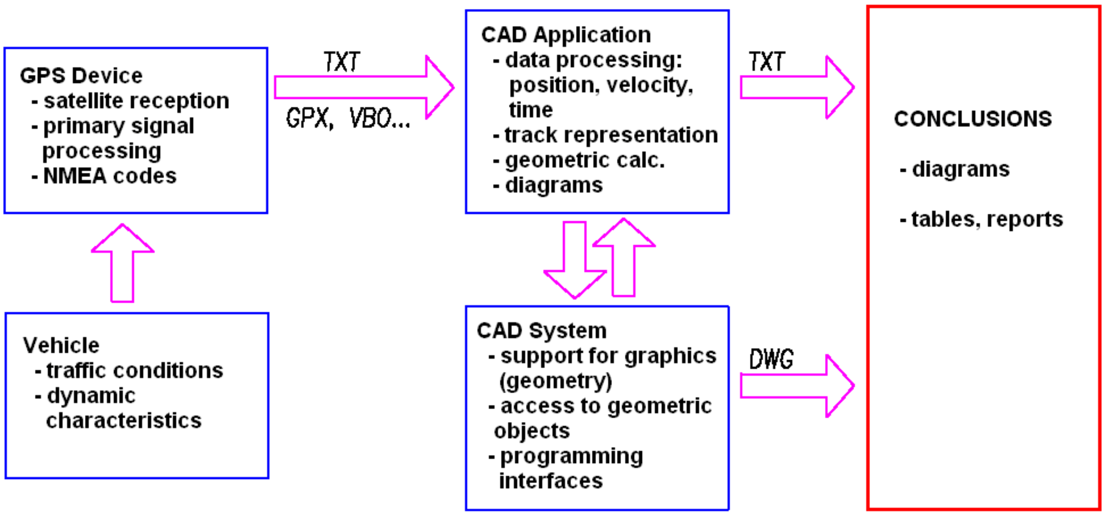


Fig. 3.23 – Flow diagram of data acquisition and processing

The process of data acquisition and processing is shown schematically in *Fig. 3.23*. Data collection phase is contained in two blocks in the first column and consists in the instrumented vehicle method, using the devices and software described in sections 3.1.1 and 4.2 respectively. The pre-processing phase is contained in the blocks in the middle column and is realized with the application presented in section 4.2.

3.2.1. Instrumented vehicle method

The preferred method for collecting data is the *instrumented vehicle* method. The instrumented vehicle is a vehicle road user with data acquisition equipment installed. Inside this book it is considered that the equipment installed on vehicle records only those data related to his behaviour, not also for other vehicles.

It is important that the instrumented vehicle to have a similar behaviour of the majority of vehicles in traffic, so that the recorded data to be representative as possible. Data are taken from the GPS receivers, but can be read also using the OBD-II interface or some sensors (engine speed sensors or others). The basic data collected from the GPS receiver are: position, time and velocity.

Some of the advantages of the instrumented vehicle method are as follows:

- There is not necessary equipment installed on various places on the route (radar, inductive loop, video camera);
- Speed is recorded for the entire route length, not only in a single point (like when using external speed sensors);
- It can be ascertained also the acceleration, since the record is continuous in time;
- The time needed to collect data on different routes is shorter;
- It can be obtained an accurate velocity field on the travelled routes.

For collecting traffic data the *instrumented vehicle* method can be extended, to the *moving observer* method. For that, a person travelling with the instrumented vehicle as passenger, can notice the number of oncoming vehicles, the number of vehicles overtaken, and the number of vehicles overtaking the observer (the instrumented vehicle). Based on the instrumented vehicle velocity, and on the mean velocity of the oncoming vehicles (estimated) and on the length of the travelled route, it can be determined the traffic flow density on that road segment [35].

The moving observer method has the following advantages:

- in the same time, beside the information about velocity can be collected data about the traffic flow (this is useful especially when studying the relation between these parameters);
- it can be determined in the same time the route length and the travel time and also the traffic flow intensity and the mean speed of the flow;
- the number of persons/hour needed for counting vehicles is smaller than in case of stationary measurements, in different points on route, being a more efficient (cost-effective) method;
- if needed, can be registered additional information, such as the places and causes of the delays.

There are also some disadvantages of the moving observer method:

- the quality of data is influenced by the number of intersections with adjacent streets;
- the measurement accuracy is affected by the fluctuations in traffic;
- in case of reduced traffic volumes are needed more tests and the method is no more such cost-effective.

The preparation of instrumented vehicle consists in installation of data acquisition equipment. In case of GPS devices, depending by their type, the most complex installation procedure consists in placing the external antenna on the roof of vehicle, connecting the computer and starting the devices and software applications.

Data acquisition is realized by driving the vehicles (repeatedly) along the chosen

4. SOLUTIONS IN THE STUDY OF VEHICLE BEHAVIOUR

In this chapter are presented some solutions identified and proposed for studying the vehicle's behaviour, as follows:

- an original data acquisition system, based on GPS;
- an application that runs under AutoCAD, for processing the acquired data, independently by the GPS receiver used;
- experiments applying the methods previously presented, for collecting and processing data: acceleration and braking tests, coast down tests, with dedicated functions implemented in the CAD application;
- experiments for vehicle's in-traffic behaviour analysis, based on data collected using instrumented vehicles, and processing that data using the same CAD application;
- a driving cycle developed for a particular city (example for Braşov city), based on data collected using instrumented vehicles;
- a method and an application developed for collecting and pre-processing data as inputs for noise mapping (noise generated by the road traffic) and for post-processing the outputs of the noise mapping software.

4.1. A NEW SYSTEM FOR GPS DATA ACQUISITION

Based on the Garmin GPS 18x-5Hz receiver [67] it was developed an original tool for vehicle dynamic behaviour analysis (named DS-5). The system has two main components: hardware and software. The hardware part consists in the GPS receiver, a minicomputer (like a Tablet-PC) and the connecting interface, usually a RS232-USB adapter. The software part is a stand-alone computer program developed in Delphi programming language. The program is used for data acquisition and for storing these data as text files on the computer. Optionally, the receiver can be connected to a logger instead of a computer, to save data directly on a micro-SD card.

4.1.1. DS-5 - Hardware

The system is composed by the GPS receiver (GPS 18x-5Hz from Garmin), the interface and the minicomputer. Depending by the computer used sometimes it may be necessary to use also an inverter, for plugging the computer to the vehicle 12 volts outlet.

The GPS device (GPS 18x-5Hz) has a connecting cable terminated as bare wires and it has to be connected to a RS232 interface. The voltage is supplied through one of the computer USB ports, usually a type A connector (*Fig. 4.1*). The connection diagram is shown in *Fig. 4.2*.

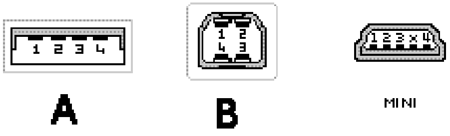


Fig. 4.1 – Pin arrangement on standard USB connectors

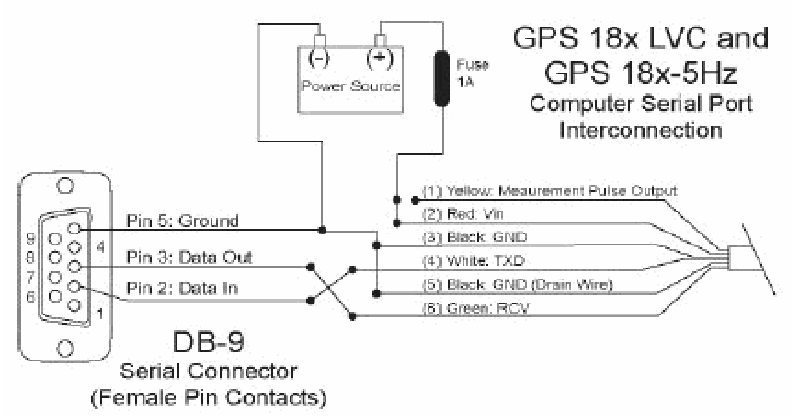


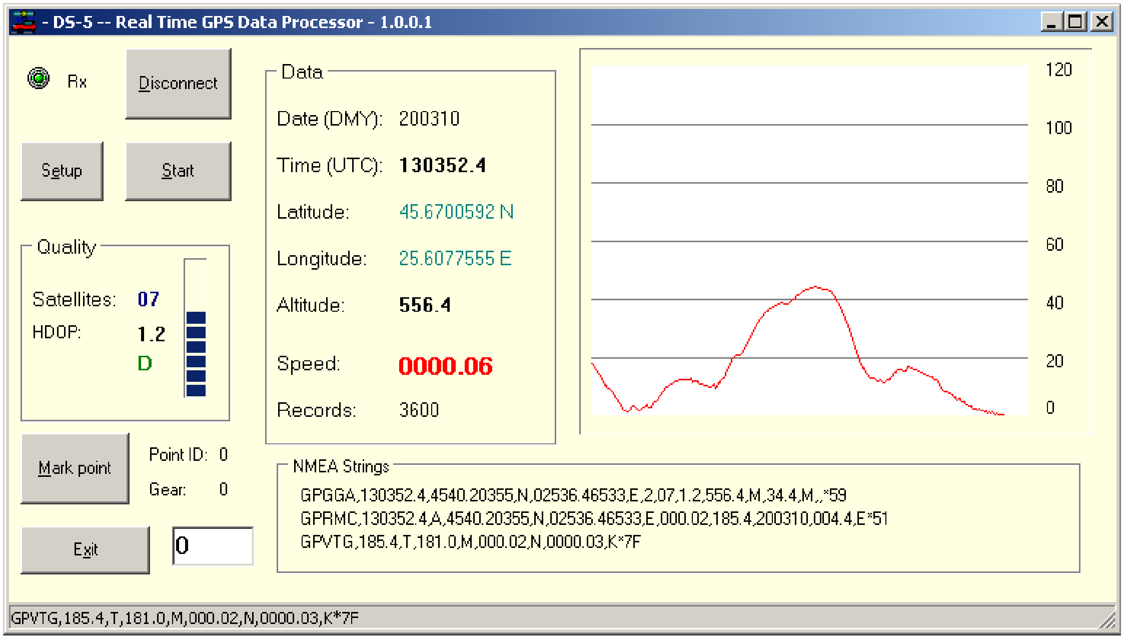
Fig. 4.2 – GPS 18x connection diagram [67]

The computer was chosen to be of small dimension and with enough performance to run the data acquisition program. The most economical solution is to use a mini-notebook from the *Asus Eee* series. These are currently the cheapest PC compatible systems; the more expensive solution is to use a Tablet-PC – this has a higher price but also some advantages (see details in *Table 4.1*).

The most important advantage of a Tablet-PC is the touchscreen, which facilitates the command inputs during tests. In case of the *Eee* computer, an efficient solution for power supply is to use an adapter 12V - 9.6V, but also is possible to use a low power inverter – it was successfully tried an inverter with a power of 75W.

4.1.2. DS-5 - Software

For acquisition, processing and saving data received with *DS-5* (based on GPS 18x-5Hz) it was developed a dedicated software application using Borland Delphi as programming environment (based on Pascal programming language). It was chosen this programming environmane because it features dedicated objects for creating flexible and easy to use user interfaces. In addition, libraries are available with predefined functions dedicated to various kind of applications, including serial port programming (RS-232). So it was possible to write the user interface shown in *Fig. 4.4*.



Satellites: 07
HDOP: 1.2
D

Date (DMY): 200310
Time (UTC): 130352.4
Latitude: 45.6700592 N
Longitude: 25.6077555 E
Altitude: 556.4
Speed: 0000.06
Records: 3600

Point ID: 0
Gear: 0

0

NMEA Strings

```
GPGGA,130352.4,4540.20355,N,02536.46533,E,2,07,1.2,556.4,M,34.4,M,,*59
GPRMC,130352.4,A,4540.20355,N,02536.46533,E,000.02,185.4,200310,004.4,E*51
GPVTG,185.4,T,181.0,M,000.02,N,0000.03,K*7F
```

```
GPVTG,185.4,T,181.0,M,000.02,N,0000.03,K*7F
```

Fig. 4.4 – User interface of the GPS data acquisition software

The main areas in the user interface (the main screen of the program) are:

- *Data* – display in real time the information received from the GPS sensor; when the satellite signal is not available (not enough visible satellites), the coordinates displayed in this field are taken from the internal memory of the device;
- *Quality* – information about quality of the signal, loke the number of visible satellites and *HDOP* (*Horizontal Dilution of Precision*); when differential signal is available, a „D” appear also in this area;
- *NMEA Strings* – the NMEA sequences received from the GPS sensor; these strings are displayed in order to have an additional control of the data received, especially in the software development phase;
- The graphic area (white area on the right side of the screen) – here is displayed, in real time, the speed versus time diagram, when data logging is enabled;
- Marking area (bottom-left) – includes the „*Mark point*” button; by clicking on

this button the current position is recorded and also the edit box „*GEAR*”, where it can be entered the current gear (numeric keys, from 0 to 6);

- The control area (upper-left) – in this area are placed the buttons for configuration, connecting to the GPS device and start/stop the recording; the buttons are big enough for easy use woth a Tablet-PC with touchscreen.

Data are recorded in text files, with the structure presented in appendix 6.2.2.

Fig. 4.5 – Complete acquisition system based on GPS 18x-5Hz

- 2 receivers, wiring,GPS data acquisition software installed on a mini-notebook (Asus Eee) and on a tablet-PC (Asus R2E)

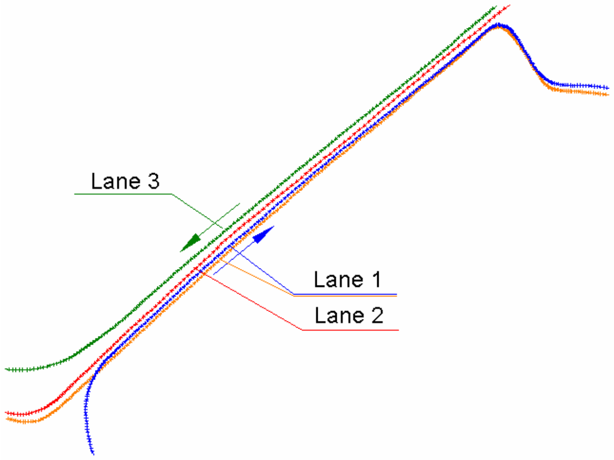


Fig. 4.6 - Tracks recorded on four vehicle passes on the same road segment, different lanes

An example that shows the accuracy of data recorded with DS-5 is in *Fig. 4.6*. Four tracks are shown, recorded when travelling with a vehicle equipped with a DS-5 system on different lanes of the same road segment. The tracks were recorded during the same day, and consequently in similar recording conditions (weather, number of visible satellites and almanac). It can be noticed that the records reflect the position of the vehicle on lanes (lanes 1 and 2 in one direction, lane 3 in the opposite direction).

4.2. CAD APPLICATION FOR GPS DATA PROCESSING

4.2.1. Program Description

The data processing application is actually a collection of AutoLisp functions. The functions may be called from a customized menu or from a customized toolbar (see examples in *Fig. 4.7*), or directly from the command prompt.

The source code is stored in some different files, *lsp* (Lisp source code) and *mns* (menu source file). The application files includes about 15.000 source lines and obviously cannot be listed here or presented in detail. However, some examples are included.

The functions can be grouped in the following categories:

- General tools (utility functions), stored in a separate file;
- Data management functions, that manage the custom data attached to geometric objects (*metadata*), stored in a sepparate file;
- Functions for reading and processing the GPS data (stored in the main source file, *gps.lsp*);
- Functions for creating and processing diagrams (also stored in the main file).

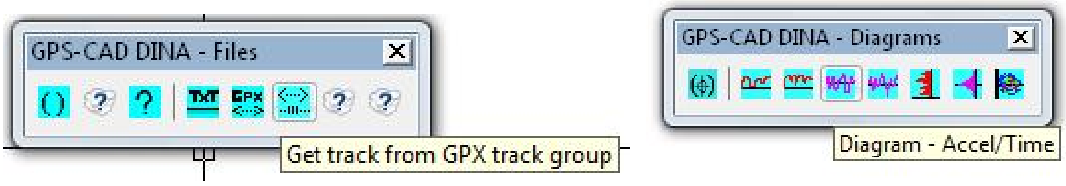


Fig. 4.7 - Toolbars included in the user interface of the data processing Lisp application

Data taken from GPS devices include information about the travelled route, draw as lines and points. Points (geometric entities of *POINT* type) have their coordinates resulted by transforming the geographic coordinates (latitude, longitude) into Cartesian coordinates *x,y* in the local projection system. The *z* coordinate can be the altitude recorded by the GPS device but, because the vertical positioning errors are relatively large, it is generally preferred that the track to be represented in plane and the altitude can be added as custom (extended) data.

Extended data (*metadata*) stored for each point include: geographic coordinates, altitude, time, velocity, longitudinal acceleration, track ID and point ID. The geographic coordinates are needed for a possible export of the track into another format (gpx), in order to view the modified tracks within other software (*MapSource, Google Earth*). Velocity is read from the GPS receiver, when is possible, or it is calculated based on position and time information.

4.3.2. Acceleration and braking tests

The acceleration and braking tests were done in order to ascertain the acceleration time and distance, and the braking time and distance. The tests were done on a straight, horizontal road. The acquisition devices used were DS-5 (with GPS 18x-5Hz sensor) and Vbox III. For short distance tests, as the acceleration and braking tests are, it is not necessary to have the track georeferenced.

There are presented below three sets of results. For the first two tests, shown in *Fig. 4.16* and *Fig. 4.17*, were used only the records made with the DS-5 device, and for the third test, shown in *Fig. 4.18*, were used the records made with both DS-5 and Vbox devices, simultaneously.

On the diagram are delimited the acceleration and braking intervals, using vertical lines. The values for the acceleration/braking time and distance are measured between these lines. The results are centralized in *Table 4.7*.

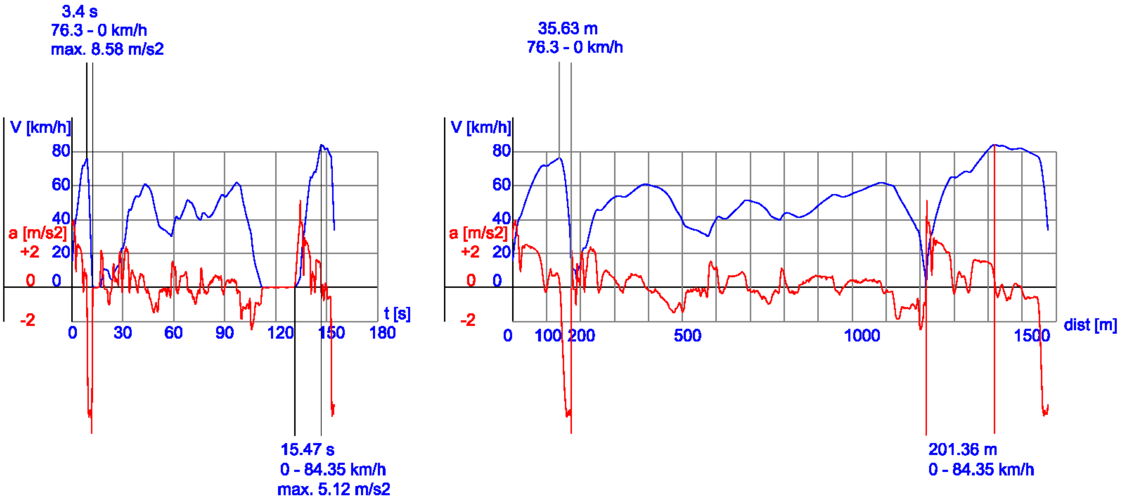


Fig. 4.16 – Acceleration and braking test – GPS 18x

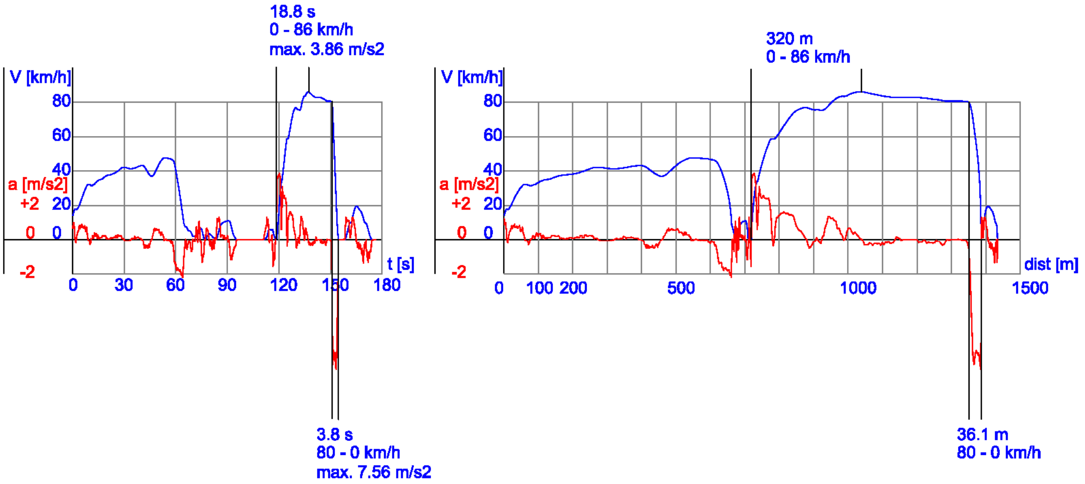


Fig. 4.17 – Acceleration and braking test – GPS 18x

4.4. ANALYSIS OF THE TRAVEL SPEED IN URBAN ENVIRONMENT

From all the routes travelled to collect data, on a period of more than two years, through Brasov City, the routes marked in *Fig. 4.36* were selected for analysis. These routes include residential streets and main arterials, the last category having a higher weight. Different vehicles were used, driven on that routes at different hours of day. The records affected by special events, like accidents or road maintenance works, were removed from the database.

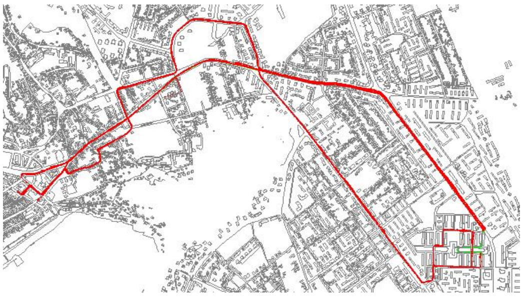


Fig. 4.36 - Routes used for collecting the kinematic data

Based on the recorded data, there were represented the diagrams of speed and acceleration variations versus time and distance. Detailed diagrams are shown in appendix 6.1. In *Fig. 4.37*, *Fig. 4.38* and *Fig. 4.39* are shown overlapped diagrams of speed variation versus time and distance, for the three routes analyzed. In each diagram, the blue curves represent the speeds and the red corves represent the accelerations; on the upper side of each figure is shown the diagram of speed and acceleration versus time, and on the bottom side is shown the diagram of speed and acceleration versus distance.

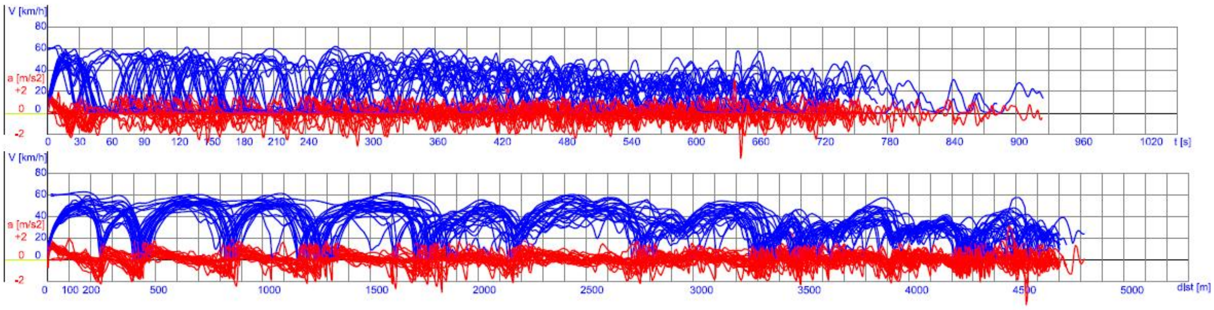


Fig. 4.37 - Speed and acceleration variation versus time and distance, route 1

The variation of speed versus distance gives a good image of driving on the respective route. It is possible to identify the stops at traffic light signals or at crosswalks, and also it can be identified the segments of arterials and of residential streets. The speed versus time diagrams look different to understand but after an automated processing it can be obtained useful results, like a particular driving cycle.

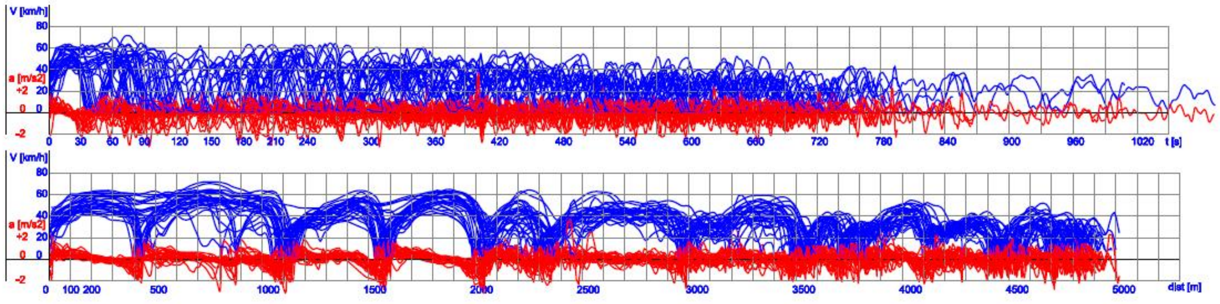


Fig. 4.38 - Speed and acceleration variation versus time and distance, route 2

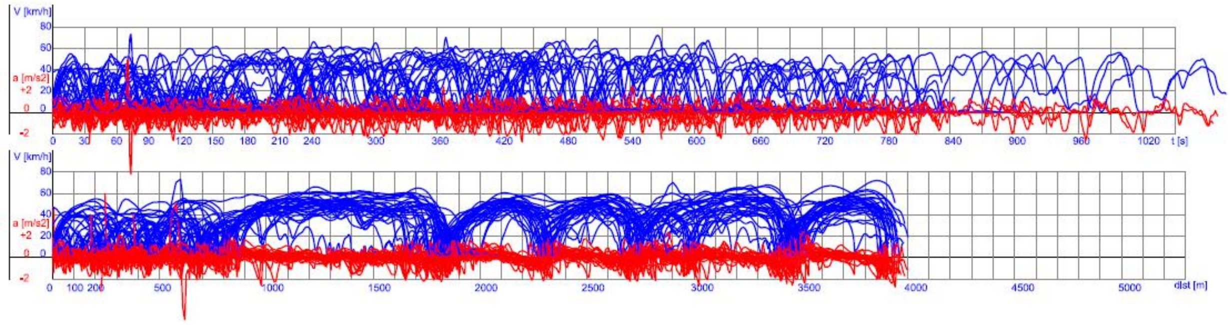


Fig. 4.39 - Speed and acceleration variation versus time and distance, route 3

From the time/speed diagrams and time/acceleration diagrams it can be ascertained the probability density of speed and acceleration respectively, as function of time, as shown in appendix 6.1, as lateral diagrams. The global probability density diagrams of speed and acceleration, for all the analyzed passings, are shown in *Fig. 4.40*.

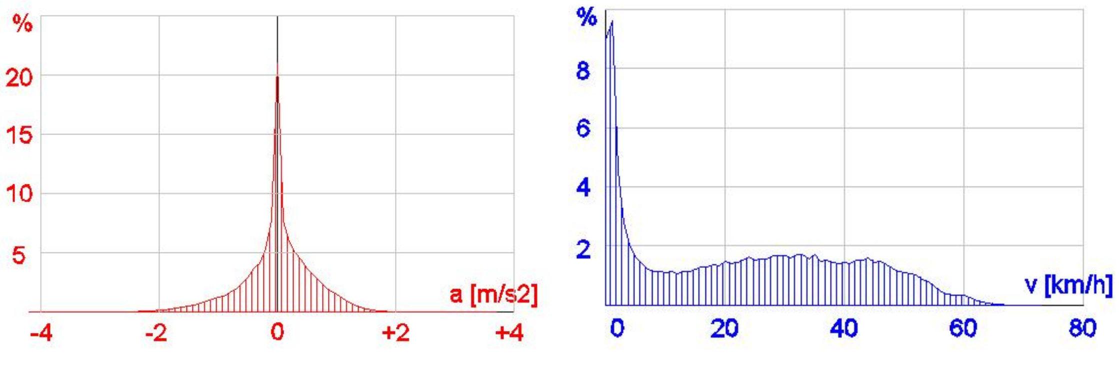


Fig. 4.40 - Probability density of speed and acceleration (as function of time)

In order to have a realistic view of speed weights, excluding the very small speeds in stop areas (signalized intersections or any other reasons), the probability density of speed and acceleration as functions of distance is presented in a similar way in *Fig. 4.41*.

From these diagrams result that the most frequently speeds are between 30 and 50 km/h. The mean deceleration (braking) is 0.45 m/s^2, and the mean acceleration (positive mean) is 0.433 m/s^2; the space-mean speed is 37.14 km/h.

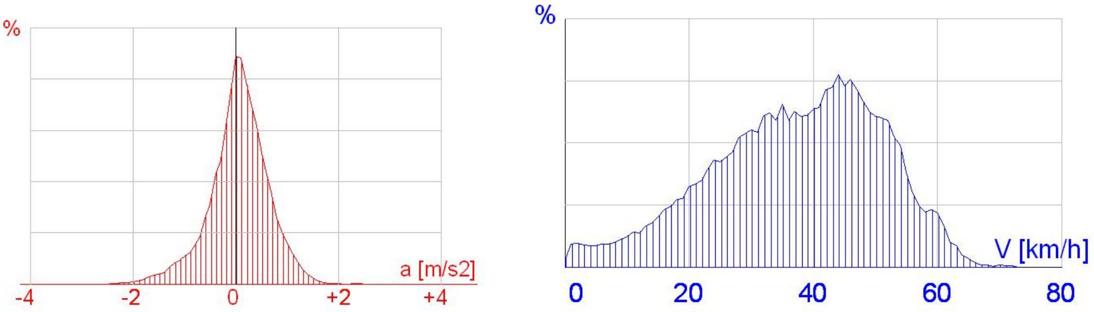


Fig. 4.41 - Probability density of speed and acceleration (as function of distance)

In appendix 6.1 are included the diagrams of probability density functions of speed and accelerations for various tracks, depending by time and distance, and also the bi-parametric probability density function for speed and acceleration.

The representation of bi-parametric probability density function for cumulated diagrams is given in *Fig. 4.42* (2D) and *Fig. 4.43* (3D). The high number of records allows a rate (cell size) of 1 km/h for speed and 0.1 m/s^2 for acceleration (for a single track a rate of 4 km/h and 0.4 m/s^2 respectively is needed, in order to fill the entire diagram area).

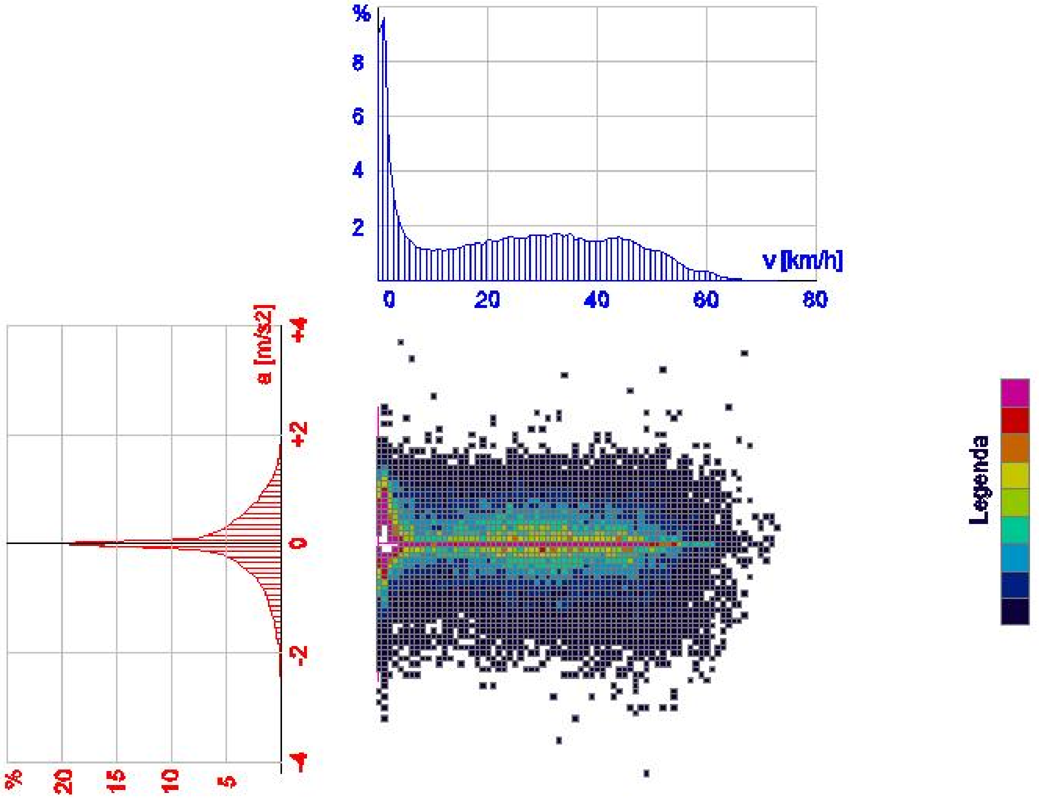


Fig. 4.42 - The bi-parametric probability density function, speed/acceleration (2D)

4.6. THE INFLUENCE OF THE TRAVEL SPEED ON THE NOISE GENERATED BY ROAD TRAFFIC

4.6.1. Estimation mode for the road traffic noise

The noise in urban areas is caused especially by the road traffic. A global image on the noise generated by a certain source (as the road traffic) in a certain area is given by the *noise map*. Based on the information included in a noise map it can be taken measures for reducing the noise level.

The noise map is realised by numerical simulation, based on input data that include the location and properties of noise sources, and also the obstacles that obstruct the noise propagation.

To date, there is not a general recognised method for evaluation of the road traffic noise. There are more calculation methods, used in different countries. In Romania it is legally accepted [74] the French method – *NMPB Routes-96* [73], which establish the dependence of the noise generated by a vehicle, by certain factors.

The factors that influence the noise generated by the road traffic are: vehicles number and their travel speed (the vehicles are divided into two categories: heavy and light vehicles, the threshold being the weight of 3,5 tonnes), the road quality, road grade and the type of the traffic flow.

The effect of noise on the environment (including people) is described using an energetical indicator that take into account both the noise level and the exposure duration. This indicator is the *equivalent sound level*, L_{Aeq}.

Since the noise on a road vary from a time to another, L_{Aeq} is a conventional measure that represents the sound level which, if should be constant for the entire reference duration, will give the same acoustic energy like the fluctuating noise of the road. The equivalent sound level is expressed in dB(A), unit that consider the sensitivity of the human ear.

The measurable parameter that is at the base of the equivalent sound level estimation is the sound pressure level. This is calculated using the equation below:

$$P_a = 20 \cdot \log_{10} \frac{P_{mas}}{P_{ref}} \qquad \text{[dB]} \qquad \textit{(eq. 4.8)}$$

where:

P_a – sound pressure level
P_{mas} – measured sound pressure
P_{ref} – reference sound pressure.

The reference sound pressure is the threshold of human hearing, which is at the sound pressure of $2x10^{-5}$ Pa.

When the noise is generated by multiple sources (like many vehicles on the same road segment) the effect of summation of all sources is calculated using the formula below:

$$P_{atot} = 10 \cdot \log_{10} \sum_{i=1}^{n} 10^{\frac{P_{ai}}{10}} \qquad \textit{(eq. 4.9)}$$

According to the national [74] and European [75] legislation requirements, the noise level must be estimated for three different periods of the day:

- day (between hours 7:00 – 19:00);
- evening (between hours 19:00 – 23:00);
- night (between hours 23:00 – 7:00).

It should be represented graphically (as noise map) the estimation results for the entire day (cumulated effect for all the three periods: day/evening/night) and separately for night. The indicators of the equivalent noise are L_{zsn} (day/evening/night) and L_n (night), respectively.

4.6.2. The influence of the speed on the noise generated by a moving vehicle

In vehicles operation it can be noticed a change in the noise level depending by the travel speed, engine speed and engine load. The French method establish in a conventional way a dependence like the nomogram presented in *Fig. 4.67* [73]. It can be observed that the noise generated is not directly proportional with the vehicle speed; it has a descendent slope up to 40-50 km/h for light vehicles and up to 50-70 km/h for heavy vehicles (for a pulsating traffic flow), then will increase – depending also by the type of the traffic flow. This is because the noise produced by automotive vehicles has several components: the engine noise, the rolling noise, the aerodynamic noise (at high speeds).

The methods for estimating the travel speeds recommended in the literature [74] are:

- Measuring the speed of the vehicles in the traffic flow using a radar device;
- Measuring the time necessary for travelling a route with a known length, then calculating the mean speed of the traffic flow;
- Ascertaining the median speed *v50* of the vehicles on the road segment; *v50* is the speed not exceeded by 50% of the vehicles, and can be determined by driving an instrumented vehicle inside the traffic flow;

- Using the legal speed limit;
- Assessing the average speed of the road traffic flor based on the experience with the road of the same type.

Depending by the chosen method, it will obtained a higher or lower accuracy of the calculation. The method are listed above sorted by accuracy (descending). The more accurate methods involve also higher costs.

In order to obtain the average speed on an urban route with a known length, the simplest method is to measure the time needed for travelling along that route. But the travel speed can vary greatly during the journey, especially if the route length is large. Furthermore, the level of the generated noise depends by the vehicle acceleration.

In these conditions, it is preferred to split any urban route into segments, so that each segment will have his own distance/speed characteristic; the main criteria for segmentation is to not have changes in the traffic flow caused by intersections, restrictions or road conformation.

For a road segment the calculation of the equivalent noise level can be calculated based on the nomogram shown in *Fig. 4.67* and on the equations presented above (a logarithmic sumation). The nomogram is defined in the French standard XPS 31-133 [73], [74] and represents the equivalent noise level L_{eq} (1h) in dB(A), called also noise emission. From the same nomogram results that the noise emission depends by vehicle speed, traffic flow and the longitudinal profile of the road.

It is possible to define an AutoLisp function to calculate the logarithmic sum as follows:

```
(defun sumlog(lista)
   ;;logaritmic sum of the list items
   (setq sum 0)
   (foreach elem lista
      (progn
        (setq sum (+ sum (expt 10.0 (/ elem 10.0))))
       )) ;;foreach
   (setq sum (* 10 (/ (log sum) (log 10.0))))
   ) ;;defun sumlog
```

In *Fig. 4.67* it can be observed that in case of a pulsating traffic flow on a horizontal road (the most frequent situation in urban areas) the minimum level of the equivalent noise generated by a light vehicle is 31 dB(A) for travel speeds between 40 km/h and 55 km/h. Then, when the speed increases to about 60 km/h the equivalent noise generated by the same vehicle rise to 32 dB(A), and at 80 km/h the equivalent noise reach 35 dB(A).

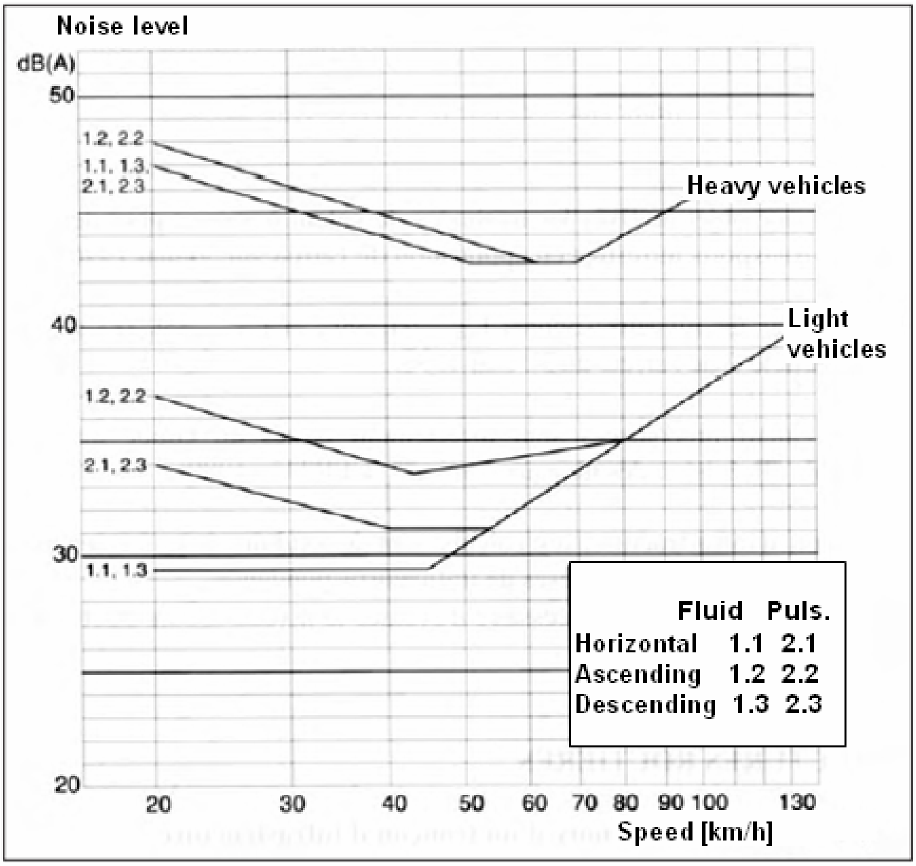


Fig. 4.67 – The noise generated by an automotive vehicle as function of speed

Assuming that on the analyzed road segment the hourly traffic volume is of 100 light vehicles, by logarithmic summation is obtained:

- For an average speed of 40 – 55 km/h: 51.0 dB(A);
- For an average speed of 60 km/h: 52.0 dB(A);
- For an average speed of 80 km/h: 55.0 dB(A).

By doubling the number of vehicles that travel the same road section in one hour means also doubling the power of the noise sources, which leads to an increase with 3 dB(A), and the following noise levels are obtained:

- For an average speed of 40 – 55 km/h: 54.0 dB(A);
- For an average speed of 60 km/h: 55.0 dB(A);
- For an average speed of 80 km/h: 58.0 dB(A).

Since doubling the traffic volume leads to an increase of the noise level with 3 dB(A), the reasoning can be extended: for 400 vehicles per hour are obtained the values of 57, 58 and 61 dB(A), for 800 vehicles per hour are obtained the values of 60, 61 and 64 dB(A) and so on in the same progression.

In case of the cities, reducing of the legal speed limit from 50 km/h to 40 km/h, in the same traffic conditions (pulsating flow) will have no effect, but reducing the number of vehicles to a half will lead to a decrease of the equivalent noise level with 3 dB(A). On the other hand, the real values of the average speed in the cities are frequently below 30 km/h, and the measures for rising this value over 40 km/h (maintaining the local limitations) will lead to a decrease of the noise level (1.5 – 2 dB (A), according to the nomogram in *Fig. 4.67*). An explanation is that the increase of noise level generated by an increasing in speed is compensated by the reducing of the exposure time. On the other hand, a lower average speed in a pulsating traffic flow involves multiple braking and accelerations, which are noise generators.

For an extended area, like a highway, a national road or a city, the calculation of the noise generated by the road traffic can be done using the general scheme shown in *Fig. 4.68*.

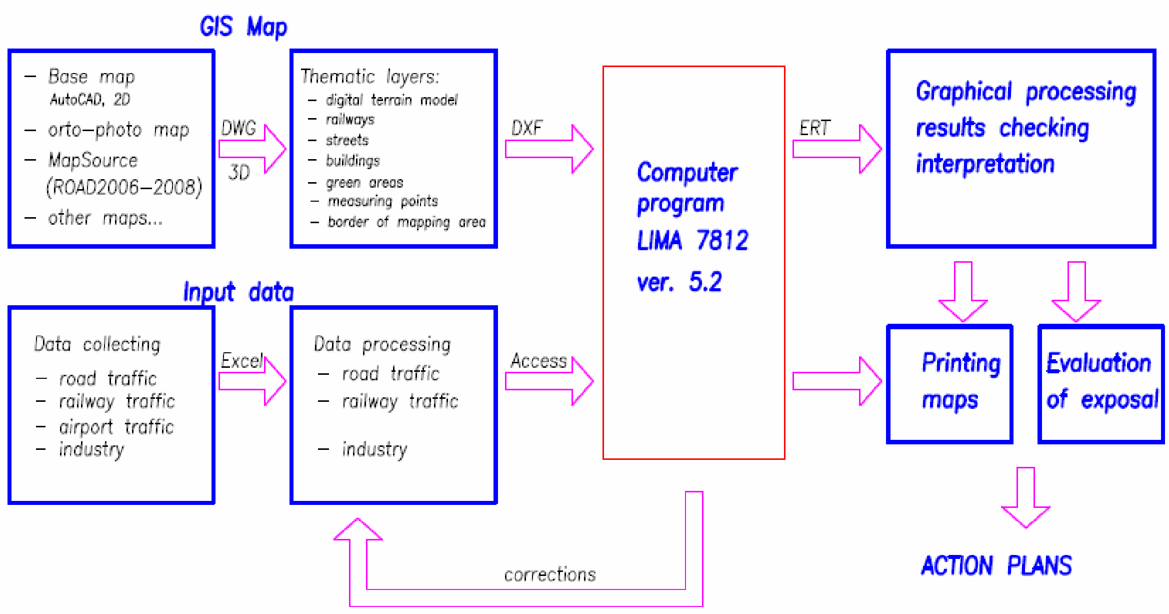


Fig. 4.68 – A general scheme for determining the noise level

There can be observed three distinctive steps of the process: collecting and preparing the input data (pre-processing), the actual calculation (in this example using Lima 7812 software) and processing the results (post-processing).

The pre- and post-processing steps can be realised using CAD applications. In addition, the input data can be collected partially using the instrumented vehicle method, described in section 3.2.2.

4.6.3. Input data – collecting and processing

The input data needed for evaluation of the road traffic noise are those data that describe the road traffic flows. More exactly, it is necessary to determine the traffic volumes (number of vehicles) and vehicles speed on each road segment in the studied area.