PART A

Theory & Fundamentals

c1 Background

> *"The true wisdom lies mainly in knowing how to teach to people with the ability to seem that you're not teaching at all, proposing things that people do not know as if they had just forgotten, proposing them in the language they know, clear and relaxed."*
>
> *A. Pope*

1.1 About road safety & design

Road safety is a shared responsibility. Reducing risk in the road traffic systems requires commitment and informed decision-making by government, industry, non-governmental organizations, and international agencies. It also requires the participation of people from many different disciplines, including road engineers, motor vehicle designers, law enforcement officers, health professionals, educators, and community groups.

To get an instant overview of the current situation about road safety, it is useful to take a look at the WHO's infographics on road safety facts (2013), that is the basis for their "decade of action" [Figure 1.1]:

- nearly 1.3 million people die in road crashes each year, on average 3,287 deaths a day;
- an additional 20-50 million are injured or disabled;
- more than half of all road traffic deaths occur among young adults ages 15-44;
- road traffic crashes rank as the 9th leading cause of death and account for 2.2% of all deaths globally;
- road crashes are the leading cause of death among young people ages 15-29, and the second leading cause of death worldwide among young people ages 5-14;
- each year nearly 400,000 people under 25 die on the world's roads, on average over 1,000 a day;
- over 90% of all road fatalities occur in low and middle-income countries, which have less than half of the world's vehicles;
- road crashes cost USD $518 billion globally, costing individual countries from 1-2% of their annual GDP;
- road crashes cost low and middle-income countries USD $65 billion annually, exceeding the total amount received in developmental assistance;
- unless action is taken, road traffic injuries are predicted to become the fifth leading cause of death by 2030.

Summarizing, nearly 3,400 people die on the world's roads every day. Tens of millions of people are injured or disabled every year. Children, pedestrians, cyclists and older people are among the most vulnerable of road users. Road safety is definitely a challenge, involving many different topics and aspects, forcing to think laterally ([1]) and focusing on a big variety of factors.

Nevertheless, road traffic crashes are predictable and can be prevented. Many countries have shown sharp reductions in the number of crashes and casualties by taking various actions, including:

- raising awareness of, legislating and enforcing laws governing speed limits, alcohol impairment, seat-belt use, child restraints and safety helmets;
- formulating and implementing transport and land-use policies that promote safer and more efficient trips; encouraging the use of safer modes of travel, such as public transport; and incorporating injury prevention measures into traffic management and road design;

[1] *From Wikipedia*. Lateral thinking (Edward de Bono, 1967) is an indirect and creative approach to solve problems, using reasoning that is not immediately obvious and involving ideas that may not be obtainable by using only traditional step-by-step logic. According to de Bono, lateral thinking deliberately distances itself from standard perceptions of creativity as either "vertical" logic (the classic method for problem solving: working out the solution step-by-step from the given data) or "horizontal" imagination (having many ideas but being unconcerned with the detailed implementation of them).

- making vehicles more protective and visible for occupants, pedestrians and cyclists; using daytime running lights, high-mounted brake lights and reflective materials on cycles, carts, rickshaws and other non-motorized forms of transport.

From the "road engineering" point of view there's room for improving roads potentials, like working on alignments, visibility, pavements performance and others. To perform any mitigation action and improve the safety on existent road network, the basic information that are needed are its geometric characteristics. Road alignment identification with mobile mapping techniques, the core of this research work, can be adopted to deepen the knowledge on existent road network and actively helps any decision make process.
Currently, there's a well-documented relationship between road crashes and alignments.

Referring to the FHWA Report (1987), wider lanes, wider shoulders, greater recovery distance, lower roadside hazard rating and flatter terrains are directly related to a reduced rate of single-vehicle accidents. Some of these assumptions have been improved and deeply studied, others have been refuted: principal contributions to this continuous process are reported below.

Hadi et al. (1993) used negative binomial regression analyses to estimate the effect of cross-section design elements on total, fatality, and injury crash rates for various types of rural and urban highways at different traffic levels. They demonstrated that, depending on the highway type that was under investigation, changing lane width, median width, inside shoulder width and/or outside shoulder width (and, in particular, increasing them), it was possible to obtain an effective reduction in crashes. Furthermore, they even showed that (on four-lane urban highways) raised median is safer than the two-way left-turn lane one.

Shankar et al. (1995) explores the frequency of occurrence of highway accidents on the basis of a multivariate analysis of roadway horizontal and vertical alignment, weather, and seasonal effects. A negative binomial model of overall accident frequencies was evaluated. They uncovered various links between specific geometrical and environmental variables with crashes and accidents; some of these variables are (e.g.) the number of horizontal curves designed below a specific speed (96.5 km/h), the overall number of horizontal curves in a section, the average spacing of horizontal curves in sections, the lowest horizontal curve radius, or the maximum grade in a section, etc.

Milton & Mannering (1998) provided a statistical model of accident frequency, using binomial regression of annual accident frequency of various sections of principal arterials in Washington State. During this estimation, they isolated the effects of various highway geometric and traffic characteristics on annual accident frequency. Moreover, they demonstrated that the negative binomial regression was a powerful predictive tool.

Abdel-Aty et al. (2000) used also negative binomial modeling technique to model the frequency of accident occurrence and involvement. The variables taken into account were again AADT, horizontal curvature, lane, shoulder and median widths, urban/rural, and the section's length. In addition to geometrical variables, they used also the demographic characteristics of the driver (age and gender). The results showed that heavy traffic volume, speeding, narrow lane width, larger number of lanes, urban roadway sections, narrow shoulder width and reduced median width increase the likelihood for accident involvement. Female drivers experience more accidents than males in heavy traffic volume, reduced median width, narrow lane width, and larger number of lanes. Male drivers have greater tendency to be involved in traffic accidents while speeding. The models also indicated that young and older drivers experience more accidents than middle aged drivers in heavy traffic volume, and reduced shoulder and median widths. Younger drivers have a greater tendency of being involved in accidents on roadway curves and while speeding.

The interesting work of Greibe (2003)established a simple accident models to predict the expected number of accidents at urban junctions and road links. In this work, he generalized linear modelling techniques to relate accident frequencies to a list of explanatory variables. The models were quite effective and were capable of describing more than 60% of the systematic variation, while the models for junctions had lower values. Furthermore, he proved that the most powerful variable for all models was the traffic flow.

Research has been developed not only on variables, but also on new modelling techniques. Chang and Chen (2005) decided to explore new methodologies, like Classification and Regression Tree (CART), which is one of the most widely applied data mining techniques in business administration, industry and engineering. CART does not require any pre-defined underlying relationship between target variables and predictors and has been shown to be a powerful tool, in particular for prediction and classification problems. A CART model and a negative binomial regression model were developed to establish the empirical relationship between traffic accidents and highway geometric variables, traffic characteristics, and environmental factors. Chang and Chen also found out that CART indicates that the AADT volume and precipitation variables are the key determinants for freeway accident frequencies. By comparing the prediction performance between the CART and the negative binomial regression models, this study demonstrates that CART is a good alternative method for analyzing freeway accident frequencies.

Caliendo et al (2007) developed some crash-prediction models for a four-lane median-divided Italian motorway. They took into account several geometric variables: length, curvature, sight distance, side friction coefficient, longitudinal slope and the presence of a junction. A separate model was developed for total crashes and for fatal and injury crashes. Results demonstrated that for curves significant variables are length, curvature, and AADT, whereas for tangents they are L, AADT and junctions.

As it has been shown within the literature review, the connection between road safety and the road alignment is sufficiently strong to improve and encourage any further effort to reduce crashes and accidents due to geometry mistakes, loss of visibility, and other geometry-related problems. Unfortunately, safety improvements on existing road are not as comfortable as with new ones. Working on the existing network is often really hard, especially when the knowledge about existing geometry is really poor or non-existent. Indeed, even modelling any crash-geometry analytical relation is a real challenge when weak or none information are available about the specific existing road under investigation, like orientation, lane width, carriageway width, length of tangents, radius of curvature, slopes, etc. These kinds of situations are common in many countries where the road network has an ancient origin and history, where the the road network has been designed and built many years ago.

This problem has assumed a big relevance after the European Guideline 2008/96/CE (2008), which in Italy has been adopted by the law D.L. 35/2011 (2012).
Making a brief summary, into this document the following technical measures have been established (already compulsory for all TEN-T European Roads, from January 1st, 2016 for all roads):

- compulsory Road Safety Audits into road design stage;
- compulsory Road Safety Inspections on existing roads;
- the whole road network safety classification;
- institution of Expert Professional Teams about Road Safety.

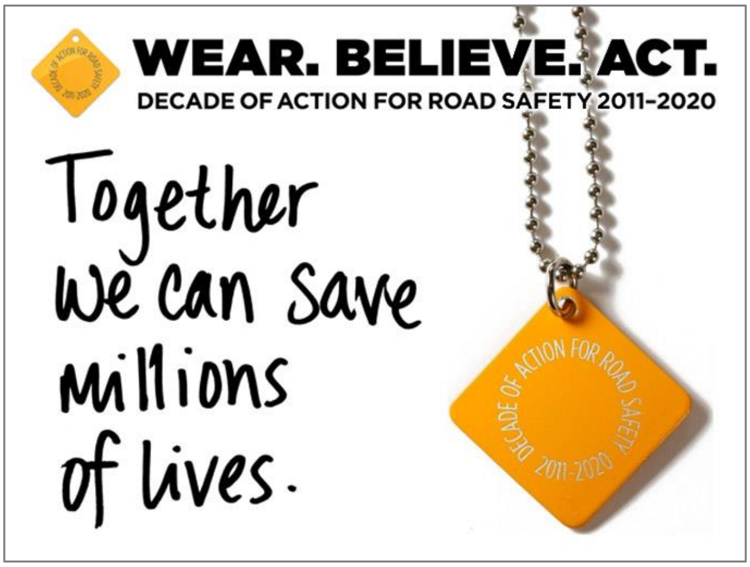

WEAR. BELIEVE. ACT.
DECADE OF ACTION FOR ROAD SAFETY 2011-2020
Together we can save millions of lives.
DECADE OF ACTION FOR ROAD SAFETY
2011-2020

Figure 1.1 - WHO logo for the "New Decade of Action for ROAD SAFETY 2011-2020"

Dealing with existent road network, Road Safety Inspections require a complete knowledge about traffic flows, pavement characteristics, and geometric properties and horizontal and vertical alignments.
Any data collection process requires a big effort in terms of time and man labor, because the most common way to get them is to perform classical surveys with traditional techniques, as well as making "on-situ" inspections and direct data acquisitions. Nevertheless, this big demand of data and information has pushed researchers to new acquisition techniques, trying to improve accuracy and sampling frequency.

In the following paragraph is exposed a brief outline of "the state of the art" regarding innovative techniques and technologies about road geometry acquisition and analysis.

1.2 Road alignments acquisition techniques

As already mentioned [1.1], big steps forward have been done in new techniques able to get spatial data information on road alignments and characteristics in various faster and smarter ways. Inside this paragraph, literature review about those recently introduced, with a specific focus on those that deal with new technologies, is included.
In particular, satellite aerial photos and mobile mapping systems will be shown, since they represent more than the 90% of recent techniques that are able to extract useful and direct information on existing road networks.

1.2.1 Satellite aerial photos and databases

GIS information and databases are quickly growing and spreading out in terms of applications and current use potential in various fields of study and work. Actually, working on GIS represents a smart way to interact with information that are also characterized by a geographic localization. GIS are databases where it is possible to store large quantities of information that can be analyzed and elaborated, and then visualized on maps.

Bhattacharya and Parui (1997) obtained satisfactory classification results using the application of backpropagation neural network for the detection of linear structures in remote-sensing images. The purpose of the approach was, firstly, to exploit the advantages of a neural network classifier over the traditional ones and secondly to avoid the strategic phases of enhancement and thresholding. Once the network is learnt, the classification scheme is in real-time.

Mena and Malpica (2005) presented in their work an efficient method for automatic road extraction in rural and semi-urban areas. They used as input the RGB bands of a satellite or aerial color image of high resolution. Getting advantage of GIS potential and generating a modular script, they ended up in an extracted road network, which is defined as a structural set of elements geometrically and topologically correct.

Imran & al. (2006) presented a global positioning system–geographic information system (GPS–GIS) based procedure for the deduction of the horizontal alignment of a road based on the path of a control vehicle. They implemented their procedure inside an ArcView extension, where the database was extracted from a 25 km section of a two-lane rural highway in eastern Ontario.

Differently from other research and studies, Easa & al. (2007) presented an approximated method to extract spiraled horizontal curves. Their search is three-dimensional based, due to the symmetry of the spiraled horizontal curve and the semiautomatic nature of the extraction process. Similar to the extraction of non-spiraled horizontal curves, the proposed method performs the search procedures in a smaller area than the image size and achieves faster computations.

The analysis of satellite images has been the research core for several years, and many are the contributions in this field. Tournaire and Paparoditis (2009) studied the road network focusing on road markings, and in particular dashed lines, proposing a new top-down approach for dashed line detection based on stochastic geometry. Özkaya

(2012) adopted an extraction approach based on Active Contour Models for 1-meter resolution gray level. Also Easa, Dong and Li (2007) worked with IKONOS 1-meter spatial resolution imagery, reaching to demonstrate that the proposed method converges in all cases and can be used for accurately establishing road horizontal curves. A different approach was proposed by Anil and Natarajan (2010) that automated the road extraction process from high resolution imagery [Figure 1.2]. A statistical region merging (SRM) was used for image segmentation, while road network was extracted using skeleton pruning methods based on contour partitioning, obtained by discrete curve evolution.

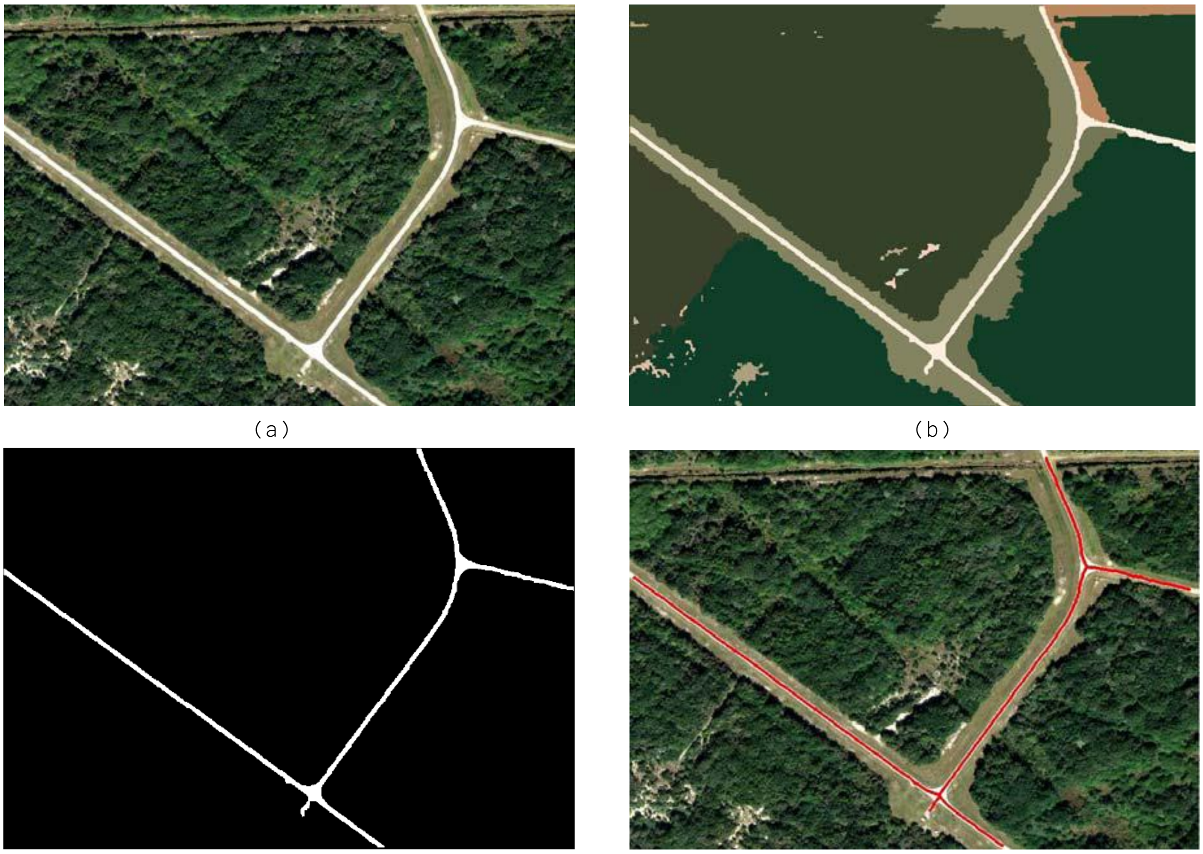


Figure 1.2 - Various steps in (Anil & Natarajan, 2010) process
(a) Original Test image-1 (b) Segmented image using SRM technique (c) Thresholded image (d) Extracted road (red) superimposed

A higher-order CRF model for road network extraction from dense urban scenes was presented in Wegner et al. (2013). A CRF formulation was proposed for road labeling, where higher-order cliques that connect sets of superpixels along straight-line segments represent the prior. Although the parameters are manually tuned for the clique sampling the results seemed very promising.

Zhao et al. (2012) proposed a method to create and/or update road maps in urban/suburban area using high-resolution satellite images. Defining “Road masks” as a mask of road pixels and “Road seeds” as directional points is possible to obtain road line extraction fusing both road mask and road seeds. They extracted main roads in high dense building area and all roads in countryside.

Chauduri & al. (2012) extracted several elements from high-resolution imagery, like roads, rivers and buildings. They developed a semi-automatic approach for road detection that exploits the properties of road segments to develop customized operators. The customized operators included directional morphological enhancement, directional segmentation and thinning. Testing the algorithm on a variety of images from IKONOS, QuickBird, CARTOSAT-2A, demonstrating that the algorithm proposed was accurate and efficient

An innovative approach was proposed by Chalambros (2014) who presented a novel framework for feature extraction and classification from images which results in the simultaneous extraction and classification of multiple feature types (surfaces, curves and joints). Combining tensor encoding, feature extraction using Gabor Jets, global optimization using Graph-Cuts, he applied this new technique in the context of road extraction from satellite images, since its characteristics makes it an ideal candidate for use in remote sensing applications where the input data varies widely.

Poz et al. (2012) proposes a semiautomatic method for 3-D road extraction in rural areas with stereoscopic satellite aerial images. They emploied a strategy based on the dynamic programming algorithm that provides a solution to the road extraction problem in the object space. In order to find road centerlines, the extraction process begins by first measuring a few seed points in one image of the stereoscopic pair and then transforming these into the object-space reference system. Experimental results showed that the proposed method is efficient and providing relatively accurate road centerlines.

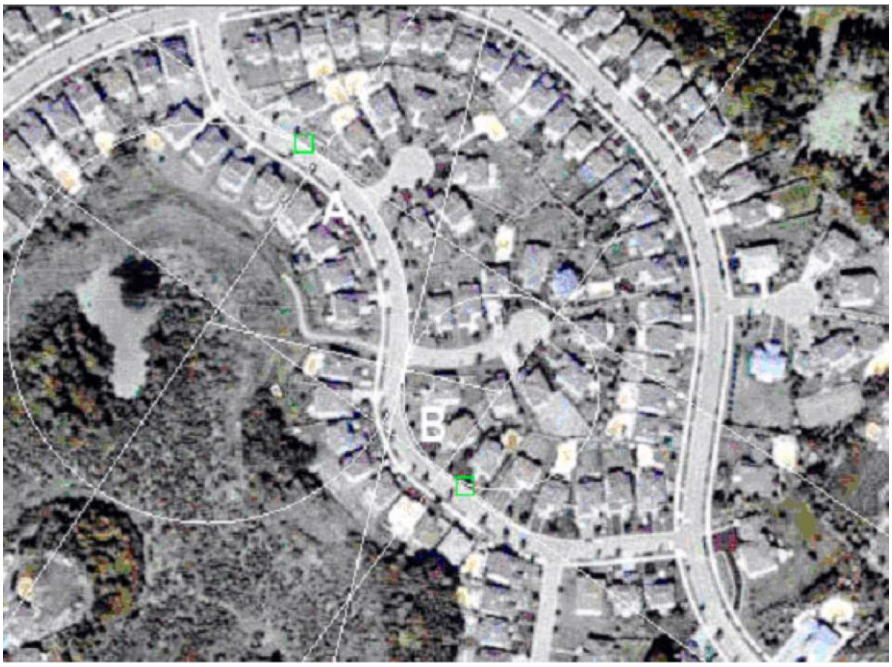


Figure 1.3 - Establishing a horizontal curve in a residential area (Easa, Dong, & Li, 2007)

Bacher and Mayer (2005), proposed an automatic road extraction technique from satellite images of rural and suburban areas. In this technique, the first step consists in the extraction of lines in all spectral channels that are then used as cues for roads to generate training areas for a subsequent automatic supervised classification. The resulting classification is finally used as an additional source for the extraction of road candidates.

Although a plethora of techniques have been proposed for the automatic or semi-automatic road extraction, the gap between the state-of-the-art and the desired goal remains wide. That is one of the reasons why other technologies have been implemented and tested to extract road alignment information, obtaining fast and enough accurate results.

1.2.2 Mobile Mapping Systems

Global Navigation Satellite Systems (GNSS) – imprecisely known as Global Positioning Systems (GPS) - is commonly adopted to find the position of fixed points on the territory, giving as result different sets of coordinates, depending of the reference system adopted in each application. Three different classes of “civil applications” for satellite systems can be highlighted:

- Different kinds of topography surveys (such as cadastral applications, designs, urban planning, etc.);
- structural deformation monitoring, mountain sides’ slow movements, etc.;
- definition, materialization and functional maintenance of a reference system, on local, regional or national scale (e.g., Regione Piemonte RTK network).

Recently, some work has been done on trying to apply these systems on tracking objects while they’re moving to capture their trajectory. This target is typical in aerial & land vehicles engineering, pushing research to the new frontier of Navigation Services. Two different techniques have been developed to find and follow a physical body along space and time, tracking its trajectory: (a) Position Fixing and (b) Dead Reckoning. Position Fixing is widely applied in GNSS and GPS technology, where the position of the receiver at each epoch (time) is completely uncorrelated with the previous one: its single position is directly obtained from the intercommunication between the receiver and the satellite; in other words, to know the position of a single point in space (and if it’s part of a trajectory, in time) it’s enough to communicate with the satellite and no information about previous epochs (points) are needed.

In the case of Dead Reckoning, instead, the positioning process is completely different, because each single position (at each single epoch) is evaluated starting from the previous one, so that it’s impossible to determine the position at a determined epoch without knowing the position of the previous one. This technique is not applied in satellite positioning, but it’s quite common with Inertial Navigation Systems (INS), that are navigation aid that use computers, motion sensors (accelerometers) and rotation sensors (gyroscopes) to continuously calculate the position, orientation, and velocity (direction and speed of movement) of a moving object without the need for external references.

(a) (b)

Figure 1.4 - Difference between "position fixing" and "dead reckoning"
(a) Each successive GPS position of physical bodies on Earth is generally single-step evaluated and doesn’t depend on previous positions
(b) In the classical maritime positioning technique the current position of the ship is calculated depending on the previous known position

Dead reckoning based sensors (like INS systems) are really useful when there’s the absolute necessity to guarantee the positioning continuity, which nowadays is not reachable with satellite systems (GNSS) that are affected by temporary signals losses (cycle slips) due to tunnels, urban canyons, deep valleys and other hostile environmental factors.

Unfortunately, INS (which are also known as Inertial Measurement Units – IMU) are not able to substitute GNSS systems completely, because they are affected by systematics errors (BIAS) that are generally not compatible with the expected precision.

Since IMU measures accelerations, a not-compensated bias produces a directly proportional error in speed and a square proportional error in position: e.g. with a bias equal to 0.04 m/s^2 the positioning error is [Figure 1.5]:

- 2 meters after 10 seconds of navigation;
- 72 meters after 1 minute of navigation;
- 1800 meters after 5 minutes of navigation.

The knowledge of the differences between these two different systems allows recognizing the leaks that can occur in case of positioning with only Position Fixing (GNSS) or Dead Reckoning (INS) technique. In the first one, problems can occur where the satellite communication is not possible or is difficult, while the second one leads to relevant errors within few seconds of acquisition.

The combination of INS high acquisition frequency and GNSS system stability generate a new set of instruments that give sufficient precise positioning with high sampling frequencies (in relation to their applications) has been designed and produced and trying to fix the cycle slips with GNSS. The accuracies obtained with such systems may vary from centimeters to meters, depending on the applications for which these sensors are designed, building technologies that are based on the different components and the extent of the errors contained in the output signal. In recent years, near to the most common construction technologies (e.g., fiber optic sensors, laser gyroscopes, mechanical accelerometers) a new sensors technology has spread on the market, called MEMS (Micro-Electro-Mechanical Sensors), characterized by a very reduced size and cost (less than 40.00 $ per sensor), pushing forward to the spread of inertial sensors based technologies and applications.
Thanks to these new efficient and relatively cheap sensors, also road engineering started to pay attention to these emerging new set of possible applications in research and road management. The chance to locate easily and rapidly a vehicle moving on the road led to many different "instrumented vehicles", generally equipped with a GNSS/INS combined positioning system coupled with a large variety of sensors.
These new "mobile wheeled capturing devices" assumed the name of Mobile Mapping Systems/Vehicles (MMS - MMV), and in next paragraphs their characteristics and applications are briefly overviewed.

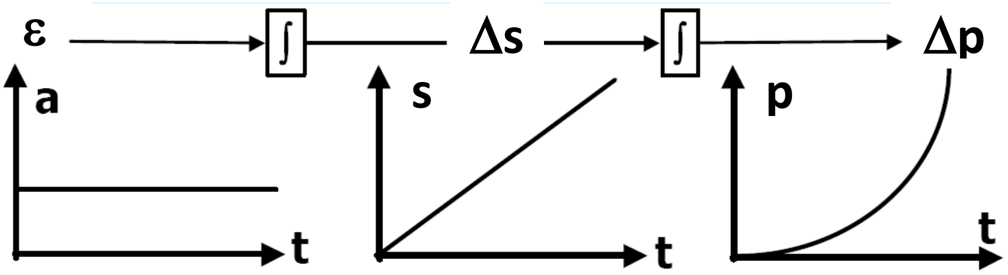


Figure 1.5 - Error behavior along iterative integration process (accelerations, speeds and positions)

1.2.2.1 Inertial Navigation Systems

Inertial Navigation Systems are generally intended the set of technological devices that are able to collect, evaluate and manage data from several inertial sensors, evaluating all navigation "states" (position, speed, acceleration, orientation, or part of them) about the monitored physical body to which is connected. An inertial sensor is a "sensitive device" able to capture and measure physical parameters (generally accelerations and angular velocity, but more often also magnetic fields) related to an inertial reference system. This data acquisition process is very independent from other sensors and with no needing to gather other information from any external device or reference (instead of GNSS systems that continuously need communication with the satellites constellation). An INS sensor is very self-sustained and when is correctly set up is able to give information independently.

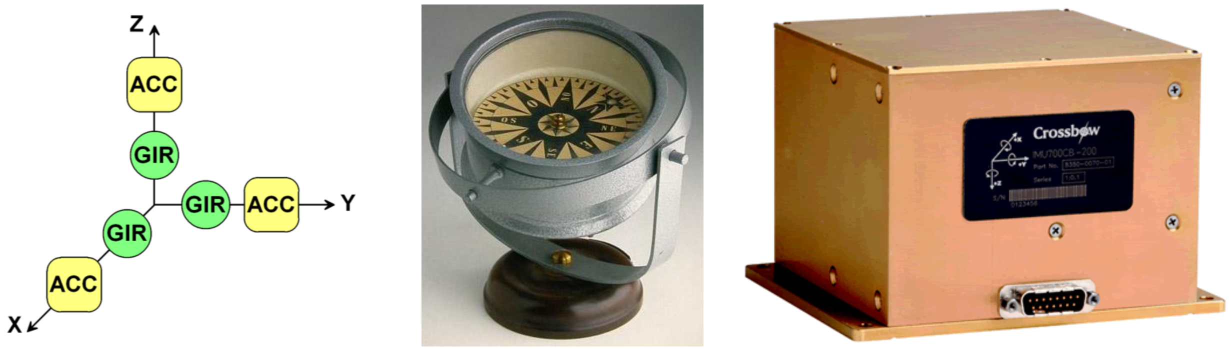
Several inertial sensors joined together in a single platform are called Inertial Measurement Systems (IMU). IMUs are composed by a first triad of gyroscopes and a second one of accelerometers, oriented on a orthogonal Cartesian system [Figure 1.6].

IMU's setup can be mainly with two different ways: gimbaled or strapdown.

A gimbal is a pivoted support that allows the rotation of an object about a single axis. A set of three gimbals, one mounted on the other with orthogonal pivot axes, may be used to allow an object mounted on the innermost gimbal to remain independent of the rotation of its support. For example, on a ship, the gyroscopes, shipboard compasses, stoves, and even drink holders typically use gimbals to keep them upright with respect to the horizon despite the ship's pitching and rolling. Adopting these properties, gimbals are then used to insulate the inertial sensors from the support, and then analyzing the mutual movement between the gimbal and the sensor is possible to evaluate variation in acceleration or angles.

Lightweight digital computers permit the system to eliminate the gimbals, creating strapdown systems, so called because their sensors are simply strapped to the vehicle. This reduces the cost, eliminates gimbal lock, removes the need for some calibrations, and increases the reliability by eliminating some of the moving parts. Angular rate sensors called rate gyros measure how the angular velocity of the vehicle changes. A strapdown system has a dynamic measurement range several hundred times that required by a gimbaled system. That is, it must integrate the vehicle's attitude changes in pitch, roll and yaw, as well as gross movements. Gimballed systems could usually do well with update rates of 50-60 Hz. However, strapdown systems normally update about 2000 Hz. The higher rate is necessary to keep the maximum angular measurement within a practical range for real rate gyros: about 4 milliradians. The data updating algorithms (direction cosines or quaternions) involved are too complex to be accurately performed except by digital electronics. However, digital computers are now so inexpensive and fast that rate gyro systems can now be practically used and mass-produced. The Apollo lunar module used a strapdown system in its backup Abort Guidance System (AGS).

Strapdown systems are nowadays commonly used in commercial and tactical applications (aircraft, ships, ROV's missiles, etc.) and they are starting to become more widespread in applications where superb accuracy is required (like submarine navigation or strategic ICBM guidance). E.g. FOG based strapdown inertial navigation systems have been selected by the UK Royal Navy for the Astute class submarine and the Queen Elizabeth class aircraft carriers.



(a) (b) (c)

Figure 1.6 – Overview of IMU gimbaled and strapdown scheme
IMU general scheme (b) Gimbaled compass (similar to gimbaled IMU) (c) Strapdown IMU

1.2.2.2 Reference frames overview

Inertial navigation is based upon the principle that is possible to determine the movement of a physical body thanks to the external forces that are acting on it. This is a very basic physical principle, expressed with the well known Newton First Law or Basic Dynamic Principle:

> *"In every material universe, the motion of a particle is determined by the action of forces*
> *whose total vanished for all times when and only when the velocity of the particle is constant.*
> *That is, a particle initially at rest or in uniform motion*
> *continues in that state unless compelled by forces to change it"*

Given as known the Second Newton Law

$$F = m \cdot a \qquad \text{Eq. 1.1}$$

It is easy to get how an INS works: knowing the mass of a physical system is possible to know its accelerations (on the orthogonal Cartesian system) simply measuring forces that are applied on the same system along the same directions.
Applying then a double integration is possible to determine the position of the system (and/or its velocity with a single integration) at every time, knowing at least a starting initial position and/or velocity:

$$V = V_0 + \int a \cdot dt \qquad \text{Eq. 1.2}$$

$$P = P_0 + \Delta P = P_0 + \iint a \cdot dt \qquad \text{Eq. 1.3}$$

where:

- **P** = Instant position of the object (P_0 the starting position)
- **V** = Instant speed of the object (V_0 the starting position)
- **a** = acceleration
- **t** = time

That's exactly what IMU does: it measures accelerations, sampled with different frequencies on a known mass, and interpretes the acceleration of the physical body to which it is connected (vehicles, aircraft, ships, etc...), evaluating than speeds and positions. Nevertheless, it is worth noting that these behaviors are valid only with inertial reference frame, and this detail is quite relevant when dealing with various reference frames. The annotation is important because in Mobile Mapping Systems there are, indeed, several different frames to which accelerations could be referred.
The main reference frames adopted are the following:

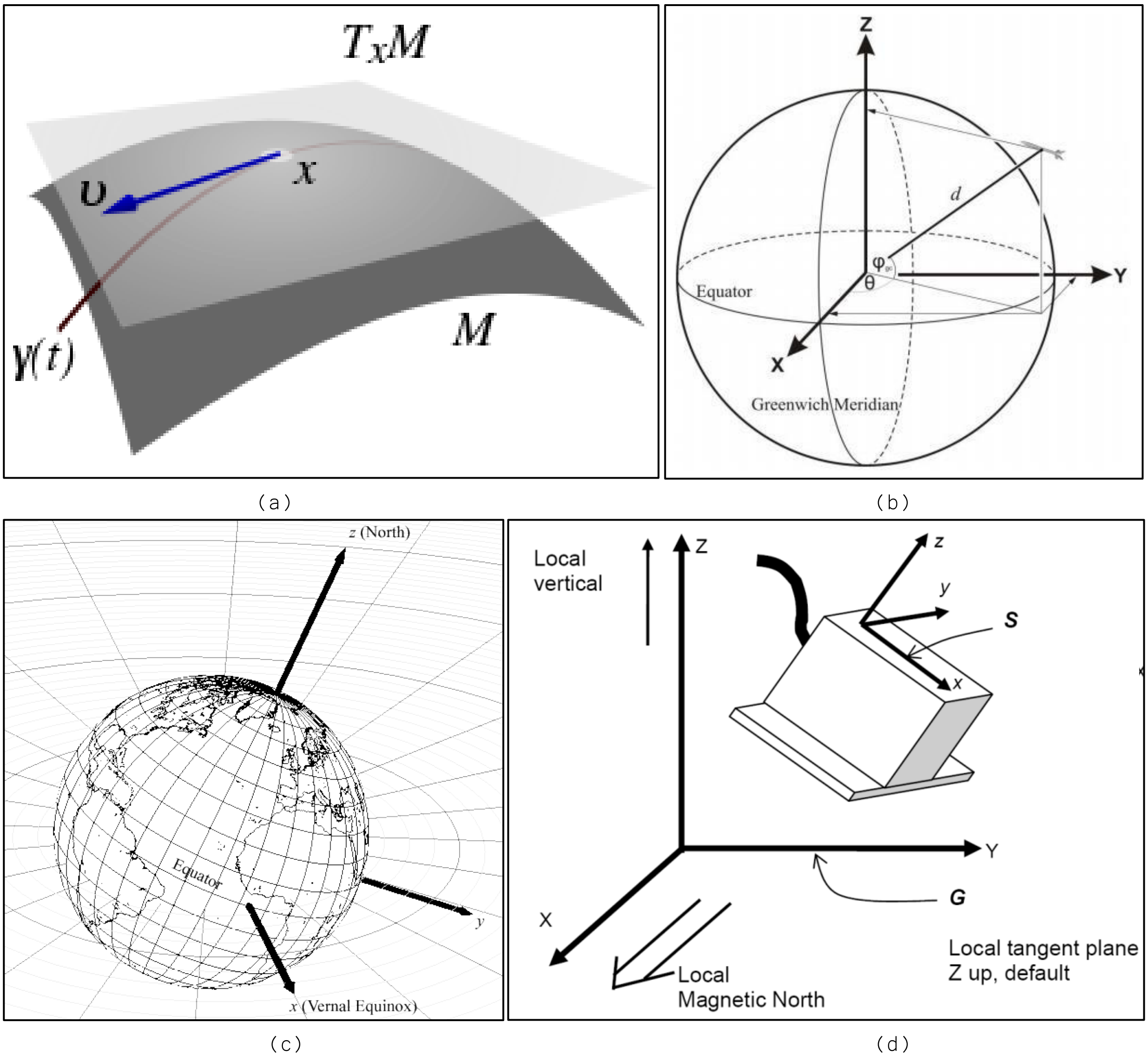
- Earth-centered frame (I-frame)
- Earth-centered & Earth-Fixed frame (E-frame or ECEF)
- Navigational frame (N-frame)
- Mobile/Body frame (B-frame)

I-frame and E-frame are both frames the center of which is located on Earth gravity center.
In the I-frame, the axis triad {x, y, z} is fixed with an astronomic reference (fixed stars), and so it *isn't fixed* with the Earth movement (neither rotation nor revolution) but it's only centered in its gravity center: this kind of frame is clearly not the best to fit any earth based positioning. [Figure 1.7-b].
A second earth-centered frame has been introduced to allow an Earth-fixed positioning, the N-frame. This frame is characterized by an axis triad {x,y,z} *fixed* with the Earth movement, and conventionally oriented: Z axis towards North Geographic Pole, X towards the Greenwich meridian (orthogonal with Z) and Y 90° counterclockwise rotated respect to X (still orthogonal with Z). This is certainly the most common reference frame adopted for navigation

purposes. [Figure 1.7-c]

The N-frame represents the orthogonal axis triad for the “local plane”. In classical topography is very common to take as reference a plane that sufficiently fits the Earth surface around the surveyed area, avoiding in this way all the necessary work to convert spherical geometry to Euclidean one. In this local frame, the triad {x, y, z} are

The last frame, the B-frame, is *fixed* with the instrument and generally is materially identified on its chassis.



(a) (b) (c) (d)

Figure 1.7 - Most common reference systems in Mobile Mapping applications

(a) Navigational frame (b) Earth-Centered & Earth-Fixed frame (c) Earth-centered frame (d) Body frame

It is always possible to switch from one system to another, mechanically (in gimbaled sensors this insulation from external movement is already guaranteed) or numerically, thanks to attitude angles of the sensors, measured from gyroscopes.

The inertial sensor samples accelerations, angle speeds with high frequencies, and it is able to process them and extract position and speed at each epoch. The IMU generally gives to the operator the possibility to set manually the output frame, so that no further conversions are necessary. The final “product” of the IMU is an output file (.txt or other property format) that contains all the information captured during the survey, for each sampling epoch. The integration between INS sensors and GNSS receiver is so that the satellite system periodically measures the position of the sensor and eventually fixes the errors produced by INS biases.

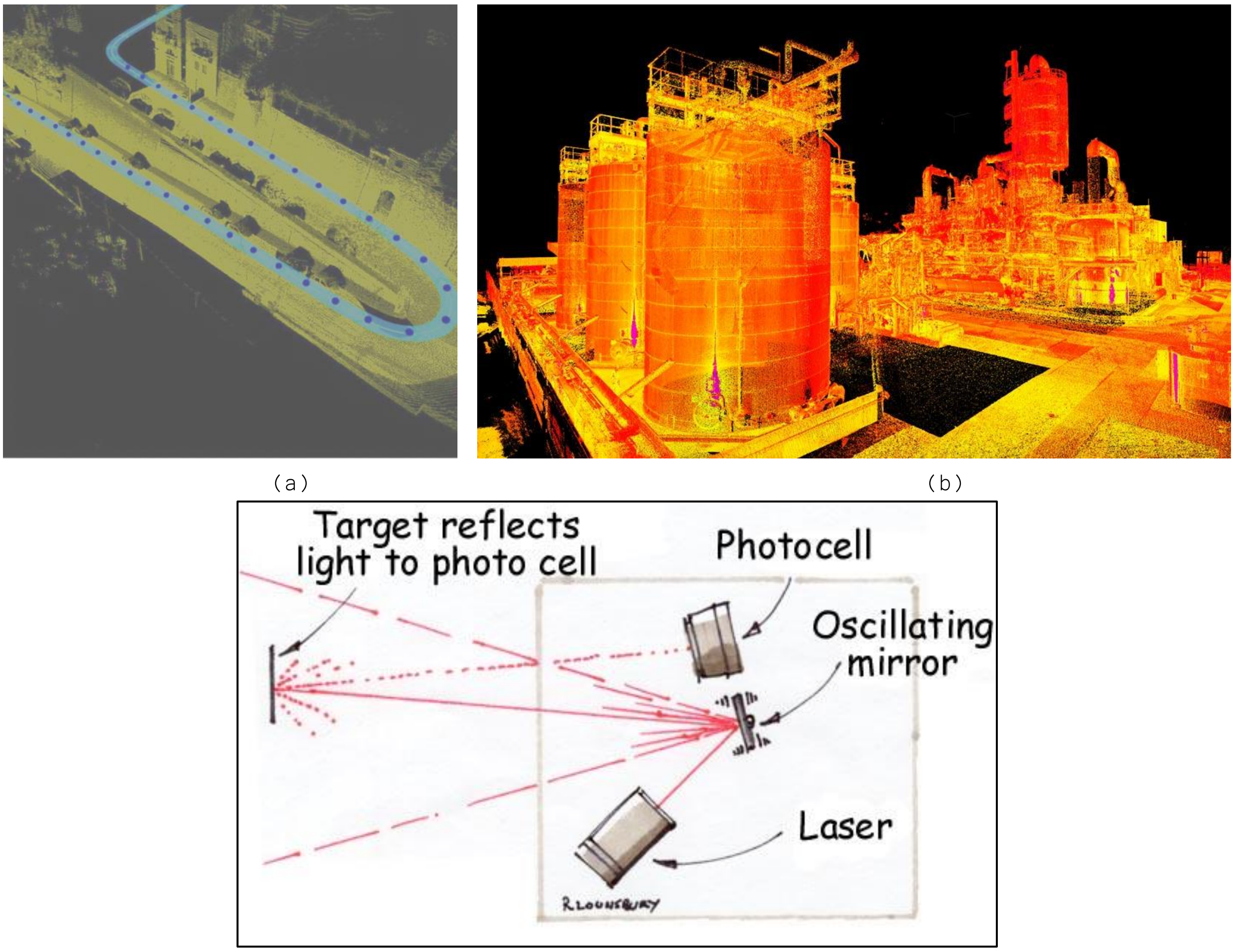
Generally, this operation is made thanks to a *Kalman Filter* [3.4]. (Manzino, 2003)

1.2.2.3 Applications

In order to have an example of what a MMV survey could look like it's useful to start from Figure 1.8. This picture shows a vehicle path, moving on the road, and the surrounding environment. However, focusing on details, it may be noticed that the environment is not just a picture or a continuous frame, but instead is a point cloud located into the space that gives to the observer qualitative information about the surrounding area. This kind of "output" is quite common by now, and it's easily obtainable from the laser scanner instrument: it's just enough to put the scanner on a vehicle (optionally with other sensors, like CCD cameras, GoPro's, proximity sensors, etc...) and the instrument start to acquire distances and angles for each returning laser beam.

A laser scanner is an instrument that produces controlled laser beams followed by a distance measurement at every pointing direction, rapidly capturing shapes of objects, buildings, and landscapes. The difference in this situation is that the laser scanner is on a moving vehicle, so each relative coordinate calculated from the scanner has to be transformed in another reference system, applying each time a correction factor due to the position and the orientation of the vehicle. Typically, it is needed to switch from the "body frame" to the "ECEF frame" [see 1.2.2.2]. That's the reason why these kinds of mobile mapping vehicles are equipped with a "positioning" system (GNSS/INSS instruments) coupled with a perceptive system (laser scanners, cameras, pollutant sensors and others): the first one gives to the second one the possibility to move from a relative "mobile" body frame to a well-known world frame; everything is kept together by an acquisition/synchronization unit and a storage.

It's easy to get advantage from this situation: it's enough to drive along a road at moderate speed to get 3D information of the surrounding environment, so that time required for survey and post-processing is sensibly reduced: that's the reason why mobile mapping systems are also called "high performance systems".



(a) (b)

(c)

Figure 1.8 - Laser scanner surveys examples and operation principles

(a) Mobile mapping laser scanning *(b) Static laser scanning* *(c) Operating principle*

Mobile mapping vehicles are used in several applications, so that they are able to provide a wide collection of geo-referred data:

- Pavement distresses geo-referred monitoring [Figure 1.9-d]
- Road Inventory applications
- Road alignments acquisition and survey
- Vertical & horizontal signposting classification
- Driver behavior
- Traffic light cycles survey and management
- Traffic data acquisition [Figure 1.9-g]
- Pollutant concentration georeferred monitoring
- Vehicle equilibrium & stability in existing road sections
- Bathymetry [Figure 1.9-e]
- Submerged pipes checking
- Oil explorations [Figure 1.9-f]
- Aerial photogrammetry
- Gravitational earth field monitoring from space [Figure 1.9-c]
- Civil Aviation: autopiloting
- Militar Aviation: real-time missile control and drones management [Figure 1.9-a]
- Autonomous-driving vehicles
- Small appliances: mower in golf fields [Figure 1.9-b]

MMV vehicles settings can vary from one to another according to their design specific aim.
Anyway, in road engineering is possible to determine four major application fields:

- road geometry identification (1),
- road pavement monitoring (2),
- driver behavior studies (3), and
- Data acquisition for virtual driving simulation applications (4).

In Italy, a decisive boost to the use and development of instrumented vehicles has been due to the introduction of the Road Inventory, according to art. 13 of the New Highway Code (1993). Vehicles manufactured for this purpose (1) are equipped with GPS, inertial sensors and video cameras in order to acquire the minimum information necessary for the formation of the cadastral database [Figure 1.10].
Some examples about this category of Italian MMVs are:

- **DAVIDE™** (Data Acquisition Vehicle with Inertial and DGPS Equipment)
- **G.I.O.V.E.** - Gruppo ELDA Ingegneria (Treviso).
- **GVS™**, (Geosoft Video Survey), Geosoft Srl (Pordenone).
- **Roadscanner**. SITECO.
- **GIGI™** (GPS Integrated Glonass and INS), GeoNetLab, Trieste University.

The basic architecture on these vehicles is the following:

- positioning system – inertial platforms and odometers, GNSS systems;
- video acquisition systems; and
- Synchronizing units.

The system is generally managed by computers or video servers, and designed according to the number of video sensors (cameras) and the amount of captured frames. Its flexibility and modularity allows the integration of additional sensor for dynamic geo-referencing of variables (e.g.: electromagnetic fields, performance of the lighting systems, etc.). Finally, road inventories databases are made in post-processing through software especially designed to reduce elaboration time. This process is composed of three main steps: (1) a relief mission with the instrumented vehicle (acquisition phase), (2) a road routes reconstruction work in the office (integration phase), and (3) the information upload through image analysis (post-processing). Furthermore, it is possible to identify two main phases: an operative filed phase and a post-processing phase in office.

Figure 1.9 - Mobile Mapping applications examples

Also abroad, many MMVs especially designed for purposes related to the acquisition of topographic data about road platforms and roadsides (alignment, lanes and shoulders width, longitudinal and transversal slopes), and road equipment have been developed. Some examples are:

- **VISAT™** (Video images, INS system, GPS Satellite system), University of Calgary, Alberta (Canada) e Geofit Inc., Laval, Québec (Canada), [Figure 1.10]
- **Photobus.** Swiss Federal Institute of Technology, Geodetic Engineering Laboratory, Lausanne (Switzerland),
- **GIPSICAM v3** (GCv3). Roads and Traffic Authority. University of New South Wales, Sydney, (Australia),
- **GPSVan™**, Ohio State University, Columbus, Ohio (USA),
- **StreetMapper 360.** Alberta's Infrastructure and Transportation, INFTRA (Canada),
- **GPSVision™**, Lambda Tech International Inc., Waukesha, Wisconsin (USA),
- **ROMDAS™** (Road Measurement Data Acquisition System), Data Collection Ltd., Motueka, (New Zealand).

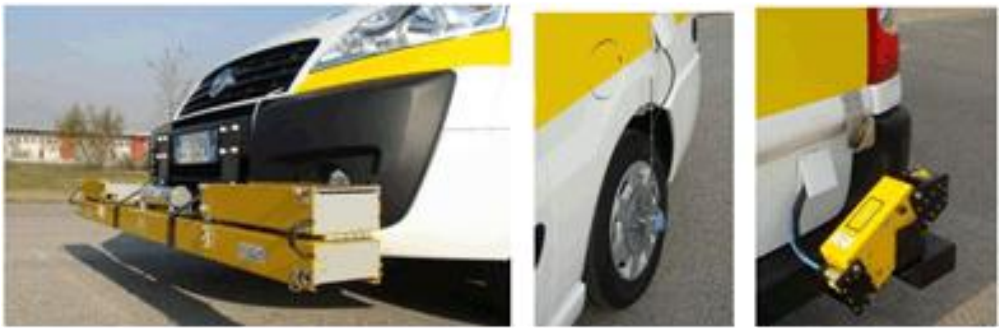


Figure 1.10 - A typical MMV equipped to identify road geometry properties

A second group of MMVs is adopted for purposes that are more closely related to the characterization of road surfaces and they generally measure quantities related to their regularity (2), lift and photometric properties:

- **ARAN™** (Automatic Road Analyzer), Roadware Corp. Paris, Ontario (Canada), [Figure 1.11]
- ROMDAS™ (Road Measurement Data Acquisition System), Data Collection Ltd., Motueka, (New Zealand),
- **Roadscanner.** SITECO (Italy), and
- **VTI - RTD** Road Surface Tester.

The Canadian ARAN is definitely the "non plus ultra" in this field. It represents the first example of a multi-sensor modular vehicle, specially designed for the analysis of road surfaces, their roughness, and the level of degradation. It hosts a variety of sensors (including reflectometers laser, ultrasonic sensors, accelerometers, GPS, gyroscopes, systems of capturing video and image) and it is able to measure up to 15 different parameters at each epoch, with high accuracy and at different speeds (from 25km/h – 15mph - up to highway speeds). With over 90 copies in circulation (including five in Italy) in about 20 countries, today the ARAN is the most used MMV, as well as the most technically and commercially relevant product in road monitoring [Figure 1.11].

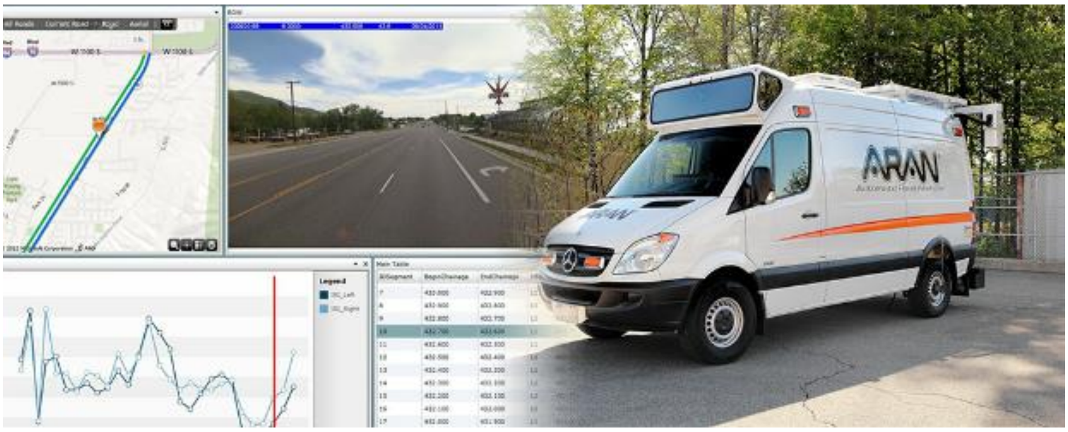


Figure 1.11 - The ARAN MMV

MMV dedicated to study driver behavior (3) are equipped with specific internal sensors and detection systems (radar proximity sensors or laser scanners), plus some others biomedical sensors (eye-trackers, heartbeat sensors, internal temperature, etc...) in order to support biomedical and psychological research activities. These are:

- **Test Vehicles Infiniti Q45 & F50**, UMTRI – Transportation Research Institute at University of Michigan (USA).
- **Computer Vision and Robotics Research Laboratory**, University of California, San Diego (USA),
- **USD research vehicle** – University of South Dakota (USA),
- **VehDAQ** (Intelligent Vehicles Lab), Federal Highways Administration (FHWA) and National Highway Transportation Safety Administration (NHTSA) (USA),
- **TRG SRIF2, EPSRC, HEFCE, TRW** and University of Southampton (GB), and
- **TTI's Instrumented Vehicle.** Texas Transportation Institute (USA).

This research sector is placed borderline with reference to specific road engineering interests, and requires non-intrusive psychological-oriented instruments. In fact, driver behavior should be investigated assuming that it is free from every kind of psychological pressures, so that is important to avoid high visible instruments. Thus, there is a huge difference from the vans showed before: sensors are almost invisible, and the driver actually feels like driving an “ordinary” vehicle keeping a natural behavior [Figure 1.12].

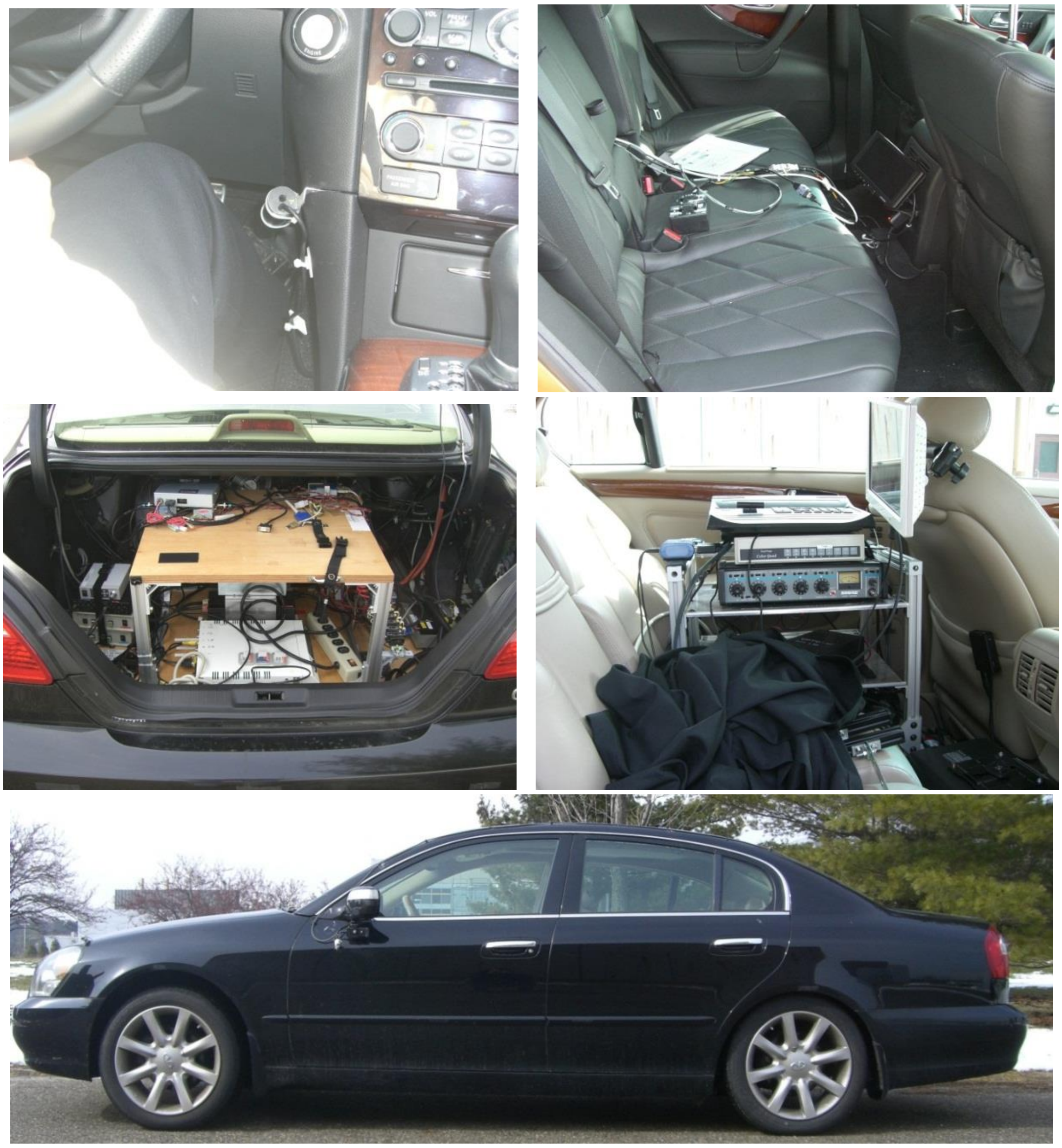


Figure 1.12 – MMV with low psychological and aesthetic impact

Finally yet importantly, there are MMVs that are focused on acquiring information about the surrounding environment (4) to recreate the driving experience (and not only that) inside a driving simulator [Figure 1.13]. Some examples are:

- **NTNU / SINTEF** (Norway),
- **ARGOS (Automobile for Research in Ergonomics and Safety).** Virginia Tech Transportation Institute, Blacksburg, VA (USA).

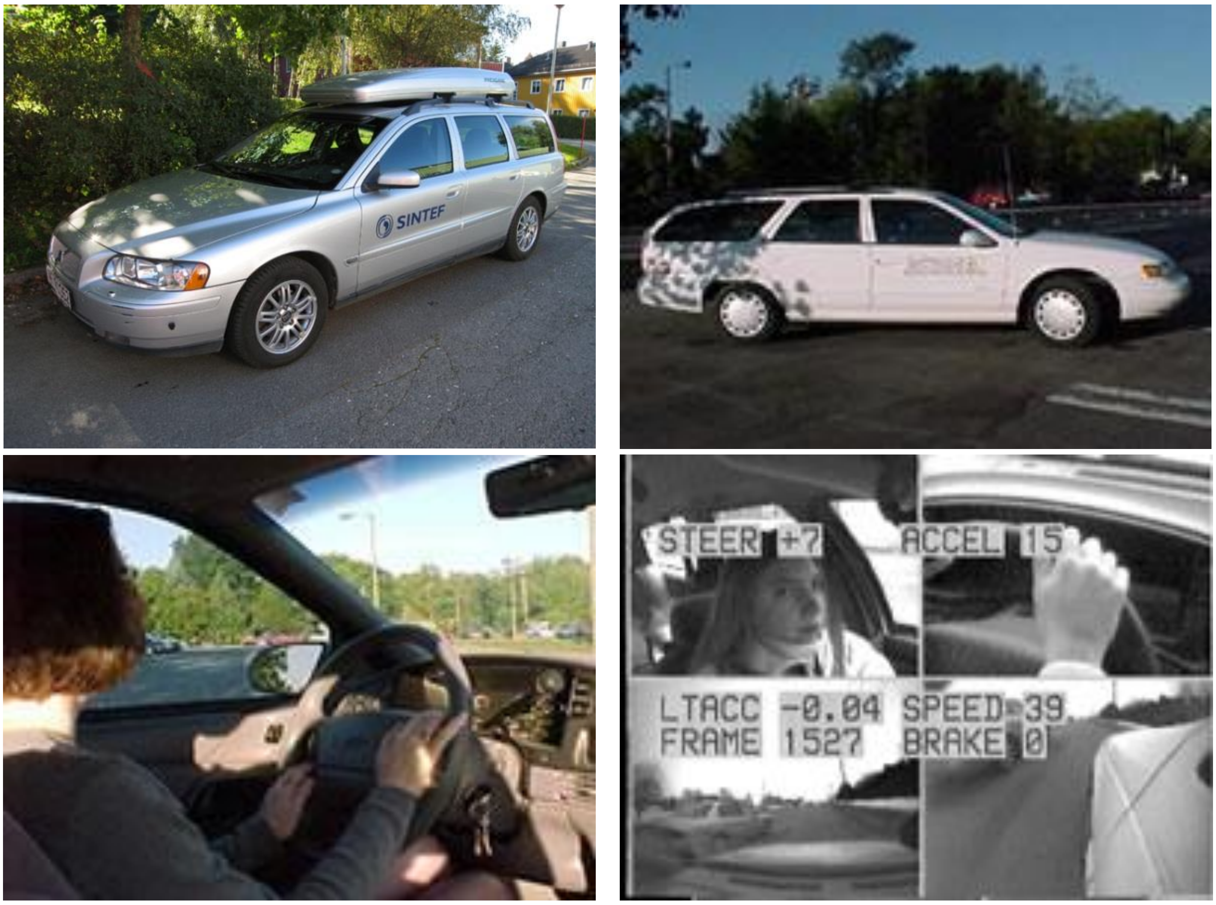


Figure 1.13 - NTNU/SINTEF MMV

c2 Alignment identification

"It can be said that the eternal mystery of the world is its comprehensibility. The fact that it is comprehensible is a miracle."
A. Einstein

2.1 The logical process: point series and clouds become roads

The survey techniques previously illustrated operate with different environments, technologies, accuracies, and final targets. However, except for that, there is a common aspect that is able to link all them each other: the output format.
Even if the output can be widely rich and various, generally it is composed by fundamental information that is a **point series.** In fact, independently from the technologies, sensors or post-processing techniques implemented, any physical element contained inside the survey is returned as a "points cloud", located in space and time. Regarding road geometries, generally they are made of "lines" in the space, composed by several points, more or less aligned one to the other.

In this research only horizontal roads geometry has been analyzed. After assuming a 2D point series, the first operation is to recognize relevant points where, along the road, there is the passage from a previous geometric element (tangents, curves or clothoids) to another one. This operation, here called "grouping", represents the first process in order to start the real geometry regression. In fact, it is fundamental that a "fitting" process, where geometrical properties of alignment are evaluated (radii, length, orientation, scale factors), may operate on points belonging to the same element.

In order to make this selection, several techniques can be adopted and tested, but the most effective ones are the **local curvature** and the **local deviation angle.**

2.2 LOCAL DEVIATION ANGLE

A road alignment can be properly assumed similar to a geometrical curve and/or a trajectory in the space. As any other curve, if it is continuous up to the second derivative (so that the continuity is always guaranteed, as well as the first and second derivative continuity) it also guarantees that is possible to evaluate, in each point of the curve and without any sudden discontinuities, the tangent line. This particular straight line is always tangent to the curve itself and can be evaluated according to the following

$$y = f'(x_0) \cdot x + f(x_0) - f'(x_0) \cdot x_0 \qquad \text{Eq. 2.1}$$

where:

- f'(x)= first derivative of the function
- x_0= general point along the function where thee tangent line is evaluated

Plotting the variation of the angular coefficient of this function (which is equivalent to its first derivative $f'(x)$), it is possible to determine any variation in "direction" of the original function or, in road engineering, the road alignment.

Actually, this is a theoretical process that is possible to be evaluated exclusively on the design project of the road, but when it's only a survey at disposal, it is necessary to discretize this concept, moving to the "local deviation angle". In Figure 2.1 it's shown an example of this concept: basically, in a discrete trajectory each single point leads to a segment that is slightly differently oriented from the previous one, generating a segment polyline. The local deviation angle is the difference in orientation between each segment of the polyline and its previous/following segment.

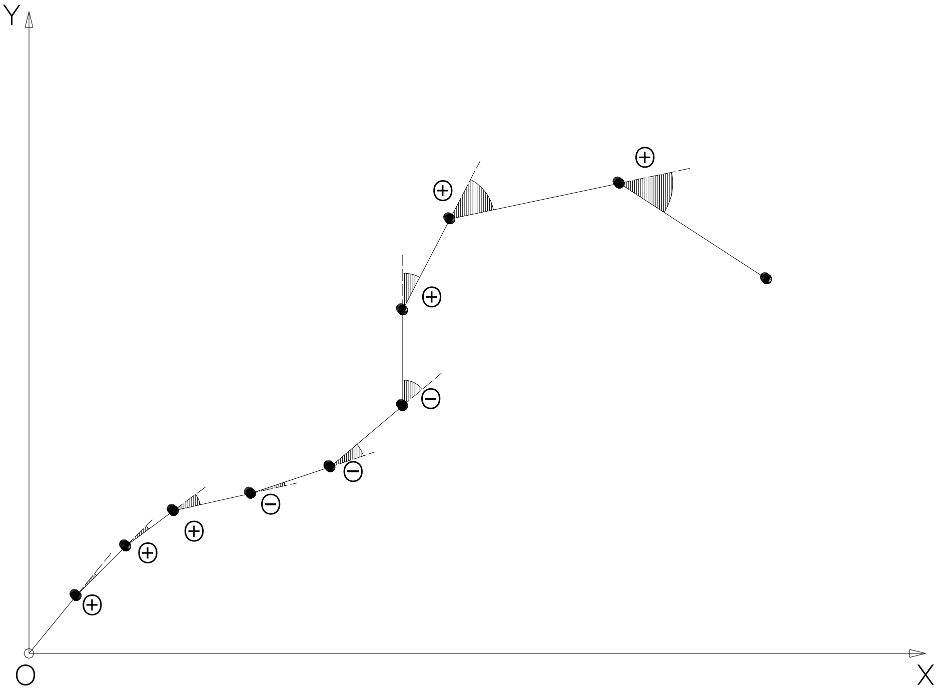


Figure 2.1 – Local deviation angle in a discrete trajectory

This specific parameter is quite good to be adopted as "flag" information that allows understanding where there is the passage from one geometric element of the road (tangents, clothoids, or curves) and the following one. There are two main ways to evaluate this specific parameter: the **triangle construction** and the **azimuth differences.**

2.2.1 The local triangle construction

With this methodology, the deviation angle is part of a triangle built along the trajectory, like the one in Figure 2.2. Calling previous epoch "0" and "1" the current one, the future "2" epoch can assume four different configurations, distinguished by the position inside the polar quadrants (I, II, III or IV). In all cases, it is possible to draw a local triangle that links together all the three epochs. Looking at this triangle, the local deviation angle (in picture θ_{2a}, θ_{2b}, θ_{2c}, θ_{2d}) represents the "external angle" in epoch 1.

According to trigonometric formulas, when all the triangle segments length are known it is possible to evaluate the triangle internal angle, corresponding to epoch 1. Referring to a generally epoch 2:

$$\begin{aligned} d_{01} &= \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2} \\ d_{02} &= \sqrt{(x_2 - x_0)^2 + (y_2 - y_0)^2} \\ d_{12} &= \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \end{aligned} \qquad \text{Eq. 2.2}$$

With the cosine equation the angle in epoch 1 is known:

$$cos\hat{1} = \frac{{d_{01}}^2 + {d_{12}}^2 - {d_{02}}^2}{2 \cdot d_{01} \cdot d_{12}} \qquad \text{Eq. 2.3}$$

It is now possible to evaluate the external angle, with the simple and well-known following equation:

$$\theta_2 = 180 - \hat{1} \qquad \text{Eq. 2.4}$$

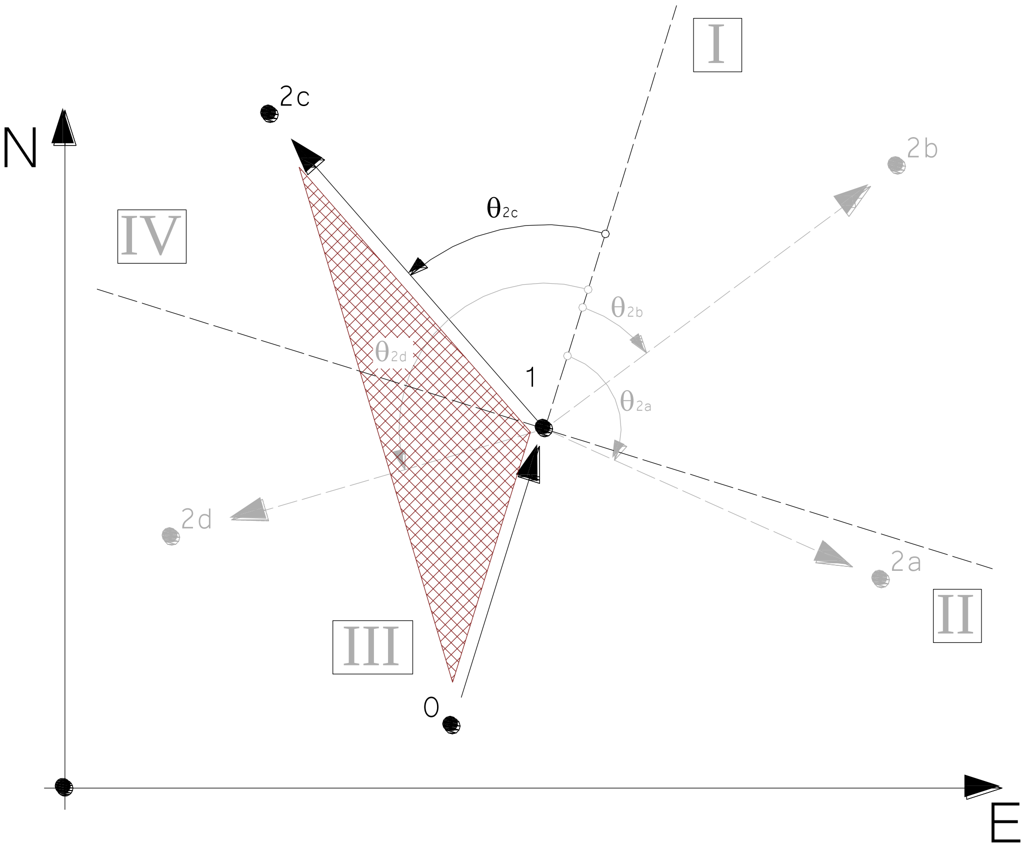


Figure 2.2 - Local triangle for deviation angle estimation
(in the picture only one of the four alternative triangles is highlighted)

2.2.2 The azimuth differences

Another way to evaluate the local deviation angle deals with north-oriented azimuth between the 3 points involved in the local analysis. With this alternative it is enough to calculate ϑ_{01}, intended of the azimuth between epoch 0 and 1, as well as ϑ_{12} between epoch 1 and 2:

$$\vartheta_{01} = \tan^{-1}\left(\frac{x_1 - x_0}{y_1 - y_0}\right)$$
$$\vartheta_{12} = \tan^{-1}\left(\frac{x_2 - x_1}{y_2 - y_1}\right) \qquad \text{Eq. 2.5}$$

Once these two azimuths are obtained, some other evaluations in terms of polar quadrant are needed. In fact, in order to adopt the deviation angle to recognize curves, a second information to the simple deviation angle, that represents the orientation, is also necessary. This second information distinguishes right-handed from left-handed curves. To assign an orientation to the deviation angle, angles are divided into positive and negative ones. Coherently with road engineering practice, right-handed curves are associated with positive deviation angles, while left-handed curves are associated with negative ones. The axis the direction of the previous epoch is assumed as reference, so in the example of Figure 2.2 is taken as reference θ_1, making evaluations on north-oriented azimuths (ϑ) between the three points:

$$if\ \ \vartheta_{12} - \vartheta_{01} < 180° \ \rightarrow\ right - handed\ curve\ \rightarrow\ \ \theta_2 = \vartheta_{12} - \vartheta_{01} \qquad \text{Eq. 2.6}$$

$$if\ \ \vartheta_{12} - \vartheta_{01} > 180° \ \rightarrow\ left - handed\ curve\ \rightarrow\ \ \theta_2 = 360° - (\vartheta_{12} - \vartheta_{01}) \qquad \text{Eq. 2.7}$$

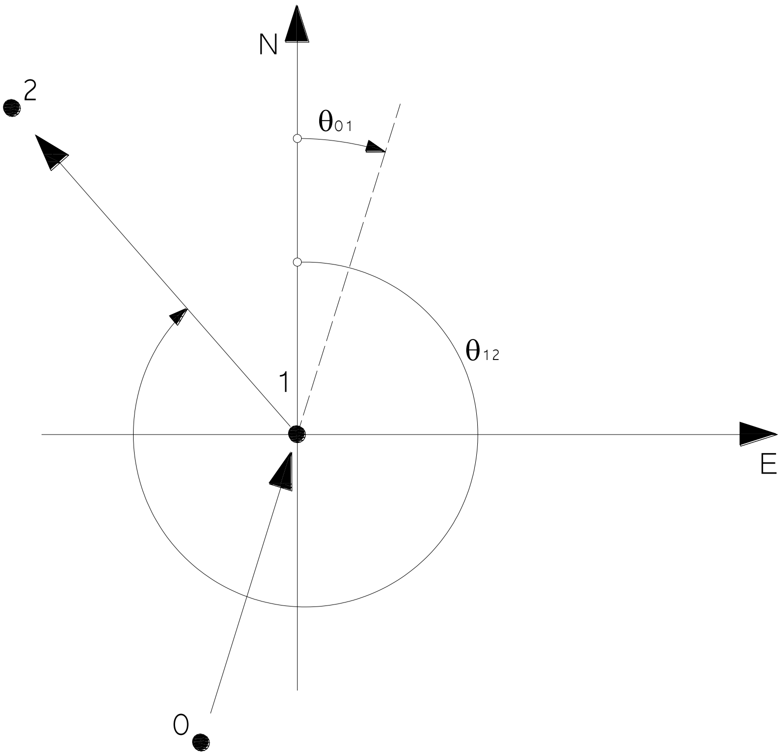


Figure 2.3 - Azimuth differences technique graphical scheme

In MatLab©, these evaluations on polar quadrants are made thanks to the function *atan2*, which returns the four-quadrant inverse tangent. This function is able to provide automatically the correction indicated above when coordinates of the points interested are known.

2.2.3 Error propagation

The two methodologies shown before provide the local deviation angle, but it is necessary to evaluate which one of them is the more suitable for this application. In fact, since all points will be affected by an error, it is preferable to choose the technique that leads to minor errors in terms of error propagation, that it is the natural mathematical tendency to produce error amplification during each elaboration. The best algorithm between many different alternatives can be intended to be the one that leads to minor error propagation.
In literature, it is possible to estimate the error propagation that occurs where an error-affected measurement is applied to calculate other derivate quantities. If "f" is a general function **and a**, **b** and **c** are independent parameters, it's possible to apply the following uncertainty propagation law:

$$s_f = \sqrt{\left(\frac{\partial f}{\partial a}\right)^2 s_a^2 + \left(\frac{\partial f}{\partial b}\right)^2 s_b^2 + \left(\frac{\partial f}{\partial c}\right)^2 s_c^2}$$

Eq. 2.8

where:

- s_f: standard deviation of the function **f, and**
- s_a, s_b, s_c: standard deviation of variable **a**, **b** and **c** respectively.

Considered this last equation, in order to establish which one is the best technique in terms of accuracy, the error propagation factor has been evaluated for both: the triangle techniques deals with the cosine equation and distances evaluation, while the azimuth differences requests azimuth evaluation and angle differences.
So that:

- Azimuths

$$\vartheta_{AB} = \tan^{-1}\left(\frac{x_B - x_A}{y_A - y_A}\right) \rightarrow basic\ equation$$

Eq. 2.9

$$\frac{\partial \vartheta_{AB}}{\partial x_B} = \cdots = \frac{y_B - y_A}{\overline{AB}^2} \quad \rightarrow first\ derivative\ on\ x_B$$

Eq. 2.10

$$\frac{\partial \vartheta_{AB}}{\partial x_A} = \cdots = -\frac{y_B - y_A}{\overline{AB}^2} \quad \rightarrow first\ derivative\ on\ x_A$$

Eq. 2.11

$$\frac{\partial \vartheta_{AB}}{\partial y_B} = \cdots = -\frac{x_B - y_A}{\overline{AB}^2} \quad \rightarrow first\ derivative\ on\ y_B$$

Eq. 2.12

$$\frac{\partial \vartheta_{AB}}{\partial y_A} = \cdots = \frac{x_B - x_A}{\overline{AB}^2} \quad \rightarrow first\ derivative\ on\ y_A$$

Eq. 2.13

- Distances

$$d_{AB} = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2} \rightarrow basic\ equation$$

Eq. 2.14

$$\frac{\partial d_{AB}}{\partial x_B} = \cdots = \frac{x_B - x_A}{\overline{AB}} \quad \rightarrow first\ derivative\ on\ x_B$$

Eq. 2.15

$$\frac{\partial d_{AB}}{\partial x_A} = \cdots = \frac{x_A - x_B}{\overline{AB}} \quad \rightarrow first\ derivative\ on\ x_A$$

Eq. 2.16

$$\frac{\partial d_{AB}}{\partial y_B} = \cdots = \frac{y_B - y_A}{\overline{AB}} \quad \rightarrow first\ derivative\ on\ y_B$$

Eq. 2.17

$$\frac{\partial d_{AB}}{\partial y_A} = \cdots = \frac{y_A - y_B}{\overline{AB}} \quad \rightarrow first\ derivative\ on\ y_A$$

Eq. 2.18

- Cosine equation

$$cos\alpha = \frac{b^2+c^2-a^2}{2bc} \qquad \text{Eq. 2.19}$$

$$\frac{\partial\alpha}{\partial b} = \cdots = \frac{-\frac{1}{c}+\frac{b^2+c^2-a^2}{2bc}}{\sqrt{1-\frac{(b^2+c^2-a^2)^2}{4b^2c^2}}} \quad \rightarrow first\ derivative\ on\ b \qquad \text{Eq. 2.20}$$

$$\frac{\partial\alpha}{\partial c} = \cdots = \frac{-\frac{1}{b}+\frac{b^2+c^2-a^2}{2bc}}{\sqrt{1-\frac{(b^2+c^2-a^2)^2}{4b^2c^2}}} \quad \rightarrow first\ derivative\ on\ c \qquad \text{Eq. 2.21}$$

$$\frac{\partial\alpha}{\partial a} = \cdots = \frac{a}{bc\cdot\sqrt{1-\frac{(b^2+c^2-a^2)^2}{4b^2c^2}}} \quad \rightarrow first\ derivative\ on\ a \qquad \text{Eq. 2.22}$$

In the triangle method it is necessary to calculate distances and the cosine equation, and to evaluate some azimuths to distinguish between right-handed and left-handed curves, while the azimuth differences method requires only azimuth evaluations and differences that are irrelevant in terms of error propagation (their partial derivatives are all equal to 1 or -1). Accordingly, the azimuth differences were adopted in this research work.

2.3 Local curvature

The second parameter adopted to recognize all different geometrical elements of the road alignment is the local curvature. This parameter represents the curvature assigned to each different point (or station) of the alignment, and it is generally the most accepted and known geometrical element that is able to give immediate information about the inspected element's nature.
Tangents, curves, and clothoids have a typical different behavior in relation to curvature that gives the immediate feeling of which kind of element is investigated.
The curvature has the following fundamental mathematical expression,

$$k = \frac{f(x)''}{(1+(f(x)')^2)^{3/2}} \qquad \text{Eq. 2.23}$$

valid for all mathematical trajectory formulations that can be derived up to the first and second level. In road geometrics, since function's gradients is quite small this equation can be transformed into the more simple and direct

$$k \approx f(x)'' \qquad \text{Eq. 2.24}$$

A further simplification can be carried out if the curvature is evaluated along a circular element. In this case, it is possible to state that

$$k = \frac{1}{R} \qquad \text{Eq. 2.25}$$

Where R is the radius of the local circumference.
It is now possible to state that:

- on circular curves the curvature is **constant**, and is equal to R^{-1};
- on tangents the curvature is **constant** too, but it's also **equal to zero**, since the local radius is **infinite**; and
- on clothoids the curvature is **linearly variable from the curvature of the previous element towards the following one.**

In addition to this numerical information, it is generally assigned a **positive** curvature for right-handed elements, while it is **negative** for left-handed ones.
The knowledge of these basic notions about curvature on different road elements gives the possibility to figure out and easily understand the typical shape of a curvature diagram of a road track with tangents, circular curves, and clothoids. Figure 2.4 provides an example of curvature diagram.

The last statement is easy to demonstrate: starting from clothoid curve fundamental equation:

$$R \cdot L = A^2 \qquad \text{Eq. 2.26}$$

and insulating **R**:

$$R = \frac{A^2}{L} \qquad \text{Eq. 2.27}$$

Remembering that the curvature is the inverse of **R**

$$k = \frac{L}{A^2} \qquad \text{Eq. 2.28}$$

Since **A** is a constant parameter of the clothoid, and remembering that the clothoid is always connected with a 2nd derivative continuity with the adjacent elements, the third statement is then demonstrated.

The information on geometrics can be collected from many different techniques such as:

- 3D space analysis,
- axis methodology,
- splines, and
- Local fitting polynomial curvature.

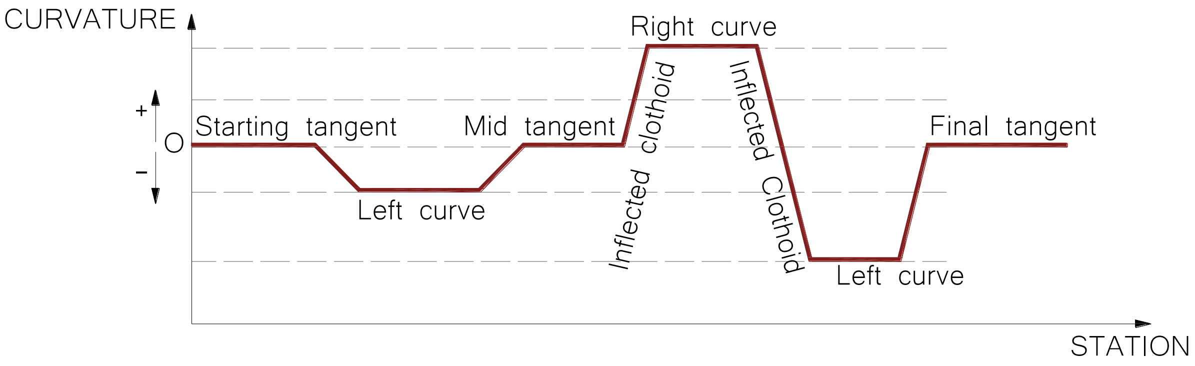


Figure 2.4 - Simple example of curvature road diagram

2.3.1 3D space analysis

The 3D approach is the only one that takes into account horizontal and vertical alignment together. In road engineering, design is split into horizontal and vertical: the vertical/horizontal coordination is a successive step that deals with visibility/perception issues. The 3D technique is characterized by a geometrical analysis that is based on three main design objects: a local **sphere,** a **plane** and a **dihedral angle.**
It is supposed to have three points in the space belonging to a trajectory to be analyzed and temporally ordered (this means that they represent different and immediate successive epoch each one to the other):

$$A = \{x_1 y_1, z_1\} \qquad B = \{x_1 y_1, z_1\} \qquad C = \{x_2 y_2, z_2\} \qquad \text{Eq. 2.29}$$

It is possible to determine the plane Π passing through these points, and it's known that it will be only one. The two vectors, **BA** and **CA**, will have the same direction of the straight lines passing from A to B, and from A to C:

$$\begin{gathered}\overrightarrow{BA} = (x_2 - x_1)\cdot\vec{\imath} + (y_2 - y_1)\cdot\vec{\jmath} + (z_2 - z_1)\cdot\vec{k} \\ \overrightarrow{CA} = (x_3 - x_1)\cdot\vec{\imath} + (y_3 - y_1)\cdot\vec{\jmath} + (z_3 - z_1)\cdot\vec{k}\end{gathered} \qquad \text{Eq. 2.30}$$

A generic point **P** belongs to the plane if

$$P(x, y, z) \in \Pi \rightarrow \overrightarrow{PA} \times \overrightarrow{BA} \wedge \overrightarrow{CA} = 0 \qquad \text{Eq. 2.31}$$

In Eq. 2.31 there is a combination of a vector product ($\overrightarrow{BA} \wedge \overrightarrow{CA}$) and a scalar product ($\overrightarrow{PA} \times \overrightarrow{BA}$). The first one gives as result a vector perpendicular to the plane Π**,** due to the nature of vector products. The second one is a scalar product between a vector on plane $\Pi\,(\overrightarrow{BA})$ and a second vector linking a generic point P to A ($\overrightarrow{PA}$). It is known by Geometry that any scalar product between two vectors is zero when the vectors are perpendicular between them. This means that if any vector linking a generic point P to A is perpendicular to $\overrightarrow{BA} \wedge \overrightarrow{CA}$, the point P belongs to plane Π**.**
With matrixes formulation, the following mathematical condition is required:

$$\begin{bmatrix} (x - x_1) & (y - y_1) & (z - z_1) \\ (x_2 - x_1) & (y_2 - y_1) & (z_2 - z_1) \\ (x_3 - x_1) & (y_3 - y_1) & (z_3 - z_1) \end{bmatrix} = 0 \qquad \text{Eq. 2.32}$$

Rearranging the formula the following equation is obtained:

$$\begin{aligned}&(x - x_1)\cdot[(y_2 - y_1)\cdot(z_3 - z_1) - (y_3 - y_1)\cdot(z_2 - z_1)] - \\ &(y - y_1)\cdot[(x_2 - x_1)\cdot(z_3 - z_1) - (x_3 - x_1)\cdot(z_2 - z_1)] + \\ &(z - z_1)\cdot[(x_2 - x_1)\cdot(y_3 - y_1) - (x_3 - x_1)\cdot(z_2 - z_1)] = 0\end{aligned} \qquad \text{Eq. 2.33}$$

Assuming that

$$\begin{aligned}\alpha &= (y_2 - y_1)\cdot(z_3 - z_1) - (y_3 - y_1)\cdot(z_2 - z_1) \\ \beta &= (x_2 - x_1)\cdot(z_3 - z_1) - (x_3 - x_1)\cdot(z_2 - z_1) \\ \gamma &= (x_2 - x_1)\cdot(y_3 - y_1) - (x_3 - x_1)\cdot(z_2 - z_1)\end{aligned} \qquad \text{Eq. 2.34}$$

the final Π implicit equation is

$$\alpha x - \beta y + \gamma z + (\beta y_1 - \alpha x_1 - \gamma z_1) = 0 \qquad \text{Eq. 2.35}$$

From which it is possible to recognize the standard formulation with the following substitutions:

$$a_p = \alpha \qquad b_p = -\beta \qquad c_p = \gamma \qquad d_p = \beta y_1 - \alpha x_1 - \gamma z_1$$ Eq. 2.36

This is a very fast and easily automatable method to find the plane.
Now, it is necessary to start looking at the sphere, superimposing that its center belongs to the plane **Π**. Starting from its generic equation

$$x^2 + y^2 + z^2 + a_s x + b_s y + c_s z + d_s = 0$$ Eq. 2.37

it is superimposed that **A**, **B** and **C** all belongs to it; it is also known that its center is on the plane **Π**, (see Eq. 2.35 and Eq. 2.36. Center coordinates are:

$$c\left(-\frac{a_s}{2}; -\frac{b_s}{2}; -\frac{c_s}{2}\right)$$ Eq. 2.38

These conditions are applicable through a system made of four equations and four unknown parameters (sphere's parameters), imposing that **A**, **B**, **C** belongs to the sphere and **c** belongs to the plane:

$$\begin{cases} {x_1}^2 + {y_1}^2 + {z_1}^2 + a_s x_1 + b_s y_1 + c_s z_1 + d_s = 0 \\ {x_2}^2 + {y_2}^2 + {z_2}^2 + a_s x_2 + b_s y_2 + c_s z_2 + d_s = 0 \\ {x_3}^2 + {y_3}^2 + {z_3}^2 + a_s x_3 + b_s y_3 + c_s z_3 + d_s = 0 \\ -a_p \frac{a_s}{2} - b_p \frac{b_s}{2} - c_p \frac{c_s}{2} + d_p = 0 \end{cases}$$ Eq. 2.39

Shortening with

$$\begin{gathered} {x_1}^2 + {y_1}^2 + {z_1}^2 = N_1 \\ {x_2}^2 + {y_2}^2 + {z_2}^2 = N_2 \\ {x_3}^2 + {y_3}^2 + {z_3}^2 = N_3 \end{gathered}$$ Eq. 2.40

the following system is obtained

$$\begin{cases} a_s x_1 + b_s y_1 + c_s z_1 + d_s = -N_1 \\ a_s x_2 + b_s y_2 + c_s z_2 + d_s = -N_2 \\ a_s x_3 + b_s y_3 + c_s z_3 + d_s = -N_3 \\ a_s\left(-\frac{a_p}{2}\right) + b_s\left(-\frac{b_p}{2}\right) + c_s\left(-\frac{c_p}{2}\right) = -d_p \end{cases}$$ Eq. 2.41

Switching to matrix formulations:

$$\begin{Bmatrix} -N_1 \\ -N_2 \\ -N_3 \\ -d_p \end{Bmatrix} = \begin{bmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ \left(-\frac{a_p}{2}\right) & \left(-\frac{b_p}{2}\right) & \left(-\frac{c_p}{2}\right) & 0 \end{bmatrix} \cdot \begin{Bmatrix} a_s \\ b_s \\ c_s \\ d_s \end{Bmatrix}$$ Eq. 2.42

Or, in a more compact form:

$$N = C \cdot S \qquad \rightarrow \qquad S = N \cdot C^{-1}$$ Eq. 2.43

By the intersection between the sphere and the plane it's obtained a circumference (the plane contains the center of the sphere) and the radius of the sphere is the same of the circumference. [Figure 2.5].

A radius is obtained on a generic plane, so it is necessary to split it up in vertical and horizontal components, . This process is possible thanks to the dihedral angle between the plane Π and the horizontal reference: computable with the two directional vectors, both perpendiculars to Π and the horizontal plane.

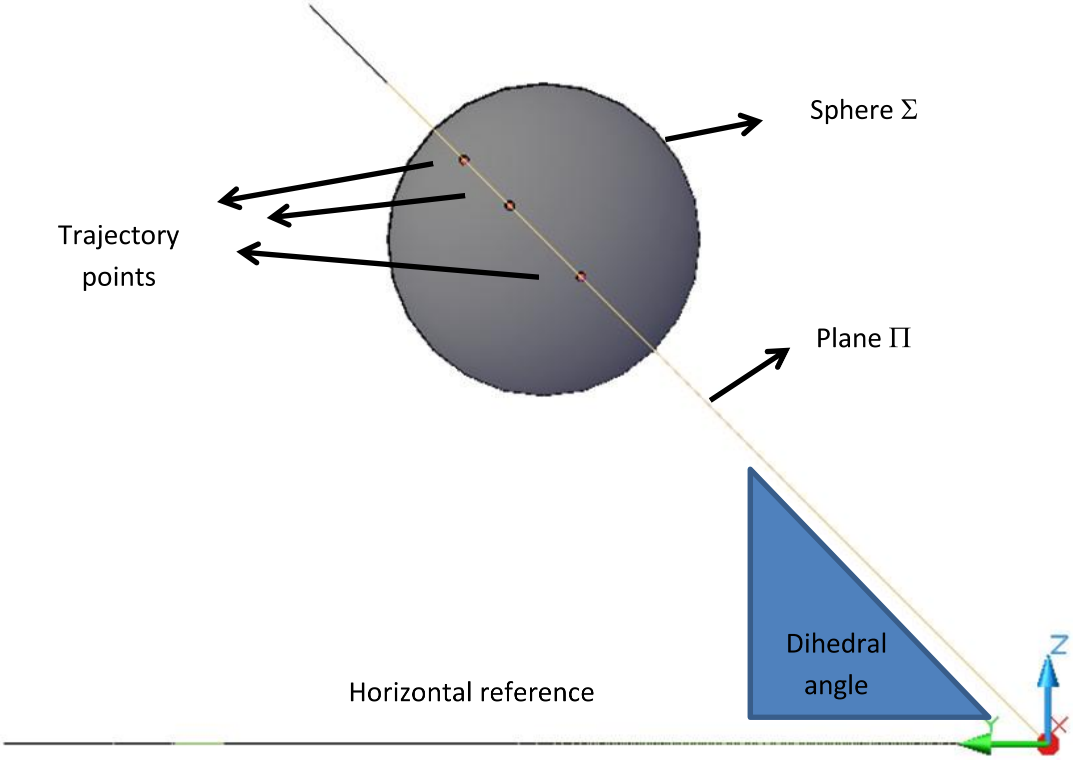


Figure 2.5 – Graphical scheme of the 3D system methodology

The mathematical formulation of this request is

$$\vec{a}(\perp \Pi) = \{a_p, b_p, c_p\} \qquad \vec{b}(\perp \mathrm{P.O.}) = \{0,0,1\} \qquad \text{Eq. 2.44}$$

and the dihedral angle is

$$\alpha = arccos\frac{\vec{a} \times \vec{b}}{|\vec{a}| \cdot |\vec{b}|} \qquad \text{Eq. 2.45}$$

The horizontal and vertical components of the curvature will be

$$\begin{aligned} R_H &= R \cdot cos\alpha \qquad \rightarrow \qquad k_H = \frac{1}{R \cdot cos\alpha} \\ R_V &= R \cdot sin\alpha \qquad \rightarrow \qquad k_V = \frac{1}{R \cdot sin\alpha} \end{aligned} \qquad \text{Eq. 2.46}$$

2.3.2 Axis methodology

This methodology is different from the previous one, because it deals exclusively with the horizontal alignment, although it is the same mathematical concept and process. With this process, it is intended that the curvature of a specific trajectory is the inverse of the radius of a circumference fitting three adjacent points. Two algorithms are capable to find the circumference:

- starting from the circumference equation or
- Adopting a geometrical process.

With the first methodology, coordinates of three consecutive points are inserted inside the generic equation of a circumference [Eq. 3.40]: working then with matrixes and mathematical systems all the parameters of the circumference are found. When a, b and c are known, the radius is immediately evaluable.
The second methodology [Figure 2.6] is sometimes adopted because from a computational point of view it is definitely easier to avoid informatics conditioning problems: often, in fact, the big magnitude of radius in road engineering and the relatively close spacing between points gives many problems in terms of fitting, getting into bad computational conditions.
Only one circumference perfectly fits a generic set of three successive points; its curvature will be associated with the central point (tipically, the second one), so that each single point is involved in three different computations: one referred to the previous point, one to the point itself and one to the following one.
Considering three successive points (#1, #2, and #3), two segments has to be identified: $\overline{12}$ and $\overline{23}$. Each segment can be characterized by an axis wich is perpendicular to it and passes through its mid-point. By Geometry, the center of the circumference is exactly the intersection between the two axes. The distance between the intersection point C and anyone of the original points will be the radius.
The first step is to find the midpoint between points 1 and 2 and between point 2 and 3:

$$x_{M_{12}} = \frac{x_1 + x_2}{2} \qquad y_{M_{12}} = \frac{y_1 + y_2}{2}$$
$$x_{M_{23}} = \frac{x_2 + x_3}{2} \qquad y_{M_{23}} = \frac{y_2 + y_3}{2} \qquad \text{Eq. 2.47}$$

Than is necessary to find the angular coefficient of the straight line $\overline{12}$ and $\overline{23}$:

$$m_{12} = \frac{y_2 - y_1}{x_2 - x_1} \qquad m_{23} = \frac{y_3 - y_2}{x_3 - x_2} \qquad \text{Eq. 2.48}$$

Since the axis are perpendicular to $\overline{12}$ and $\overline{23}$ respectively, it’s immediate to calculate their angular coefficient:

$$m_{12C} = -\frac{1}{m_{12}} \qquad m_{23C} = -\frac{1}{m_{23}} \qquad \text{Eq. 2.49}$$

The research of the center of the curve mathematically corresponds to putting in a system the equations of the two segment-axes just obtained:

$$\begin{cases} \left(y - y_{M_{12}}\right) = m_{12C} \cdot \left(x - x_{M_{12}}\right) \\ \left(y - y_{M_{23}}\right) = m_{23C} \cdot \left(x - x_{M_{23}}\right) \end{cases} \qquad \text{Eq. 2.50}$$

with some mathematical arrangements

$$\begin{cases} y = x \cdot m_{12C} + \left(y_{M_{12}} - x_{M_{12}} \cdot m_{12C}\right) \\ y = x \cdot\ m_{23C} + \left(y_{M_{23}} - x_{M_{23}} \cdot\ m_{23C}\right) \end{cases} \qquad \text{Eq. 2.51}$$

Simplifying the notations by introducing K_1 and K_2:

$$\begin{gathered} y_{M_{12}} - x_{M_{12}} \cdot m_{12C} = K_1 \\ y_{M_{23}} - x_{M_{23}} \cdot\ m_{23C} = K_2 \end{gathered} \qquad \text{Eq. 2.52}$$

the center of curvature is

$$\begin{cases} y = x \cdot m_{12C} + K_1 \\ y = x \cdot\ m_{23C} + K_2 \end{cases} \qquad \text{Eq. 2.53}$$

$$\begin{gathered} x \cdot m_{12C} + K_1 = x \cdot\ m_{23C} + K_2 \\ x(m_{12C} -\ m_{23C}) = K_2 - K_1 \end{gathered} \qquad \text{Eq. 2.54}$$

Hence:

$$x_C = \frac{K_2 - K_1}{m_{12C} -\ m_{23C}} \qquad y_C = \begin{cases} x_C \cdot m_{12C} + K_1 \\ or \\ x_C \cdot\ m_{23C} + K_2 \end{cases} \qquad \text{Eq. 2.55}$$

Eq. 2.53 combined with Eq. 2.54 are used to calculate the center coordinate. Once the center is known, the radius is simply one of the three distances from C to 1, 2 or 3, and the curvature will be its inverse.

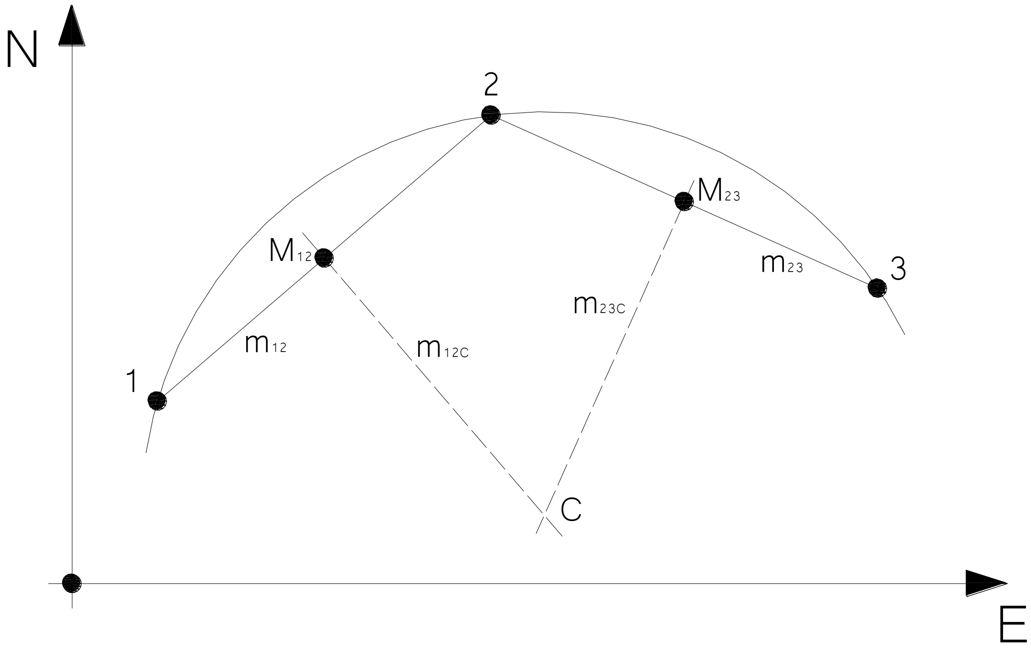


Figure 2.6 - The axis methodology scheme

2.3.3 Polynomial fitting

The third and last methodology here analyzed is the polynomial fitting..
Starting with the fundamental polynomial equation:

$$P(x) = y = c_0 + c_1 x + \cdots + c_{n-1} x^{n-1} \qquad \text{Eq. 2.56}$$

The evaluation of the curvature along a trajectory is evaluable thanks to Eq. 2.23 to it. Nevertheless, it is necessary that the equation is derivable at least two times to give information about curvature, so straight-line equations are not suitable for this purpose. Polynomial fitting can present some problems with autonomous calculation and fitting. The worst one is the coefficients matrix bad conditioning, which generally assumes the form of a Vandermonde matrix. In linear algebra, this kind of matrix is characterized by elements on the same raw (or column) ordered in geometrical progression, starting with 1.

$$V = \begin{bmatrix} 1 & \alpha_1 & {\alpha_1}^2 & \dots & {\alpha_1}^{n-1} \\ 1 & \alpha_2 & {\alpha_2}^2 & \dots & {\alpha_2}^{n-1} \\ 1 & \alpha_3 & {\alpha_3}^2 & \cdots & {\alpha_3}^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \alpha_n & {\alpha_n}^2 & \cdots & {\alpha_n}^{n-1} \end{bmatrix} \qquad \text{Eq. 2.57}$$

And, as a matter of fact, generic polynomial coefficients, which its graph passes through generic points,

$$(x_1; y_1), \dots, (x_n; y_n) \qquad \text{Eq. 2.58}$$

are solutions of the following linear system

$$\begin{bmatrix} 1 & x_1 & {x_1}^2 & \dots & {x_1}^{n-1} \\ 1 & x_2 & {x_2}^2 & \dots & {x_2}^{n-1} \\ 1 & x_3 & {x_3}^2 & \cdots & {x_3}^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & {x_n}^2 & \cdots & {x_n}^{n-1} \end{bmatrix} \cdot \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_{n-1} \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{n-1} \end{pmatrix} \qquad \text{Eq. 2.59}$$

This system is characterized by the presence of a Vandermonde matrix. This mathematical situation deals with the impossibility to adopt classical solving techniques, like LU factorization or backward substitution. When the amount of points increases also the system increase and the conditioning index is constantly increasing too.
To solve this problem the **centering and scaling technique is** usually adopted. In the case of fitting a series of points with a 3rd degree polynomial function; since there is the possibility to deal with a bad conditioned Vandermonde matrix, a change in parameters is applied, centering **x** at zero and scaling it to have unit standard deviation; assuming:

$$\mu_1 = average(x_i) \qquad \mu_2 = standard\ deviation(x_i) \qquad \text{Eq. 2.60}$$

a new parameter is introduced

$$\widehat{x_i} = \frac{x_i - \mu_1}{\mu_2} \qquad \text{Eq. 2.61}$$

All the points are scaled and centered, getting a new series of points where there are reasonable differences within coordinates units and the "Vandermonde" effect will be smaller.
Returning to the previous system, it's needed to do some math, starting from the original third degree polynomial equation (is developed to the third degree but it's possible for n-degree):

$$P(x) \to y = ax^3 + bx^2 + cx + d$$

and applying Eq. 2.61

$$P'(\hat{x}) \rightarrow y = a\hat{x}^3 + b\hat{x}^2 + c\hat{x} + d$$

$$P(x) \rightarrow y = a\left(\frac{x-\mu_1}{\mu_2}\right)^3 + b\left(\frac{x-\mu_1}{\mu_2}\right)^2 + c\left(\frac{x-\mu_1}{\mu_2}\right) + d$$

$$y = \frac{a}{{\mu_2}^3}(x-\mu_1)^3 + \frac{b}{{\mu_2}^2}(x-\mu_1)^2 + \frac{c}{\mu_2}(x-\mu_1) + d$$

$$y = \frac{a}{{\mu_2}^3}(x^3 - 3x^2\mu_1 + 3x{\mu_1}^2 - {\mu_1}^3) + \frac{b}{{\mu_2}^2}(x^2 - 2x\mu_1 + {\mu_1}^2) + \frac{c}{\mu_2}x - \frac{c\mu_1}{\mu_2} + d$$

$$y = \frac{a}{{\mu_2}^3}x^3 + \left(-\frac{3a\mu_1}{{\mu_2}^3} + \frac{b}{{\mu_2}^2}\right)x^2 + \left(\frac{3a{\mu_1}^2}{{\mu_2}^3} - \frac{2b\mu_1}{{\mu_2}^2} + \frac{c}{\mu_2}\right)x - \frac{a{\mu_1}^3}{{\mu_2}^3} + \frac{b{\mu_1}^2}{{\mu_2}^2} - \frac{c\mu_1}{\mu_2} + d$$

so that:

$$A = \frac{a}{{\mu_2}^3} \qquad B = -\frac{3a\mu_1}{{\mu_2}^3} + \frac{b}{{\mu_2}^2}$$

$$C = \frac{3a{\mu_1}^2}{{\mu_2}^3} - \frac{2b\mu_1}{{\mu_2}^2} + \frac{c}{\mu_2} \qquad D = \frac{a{\mu_1}^3}{{\mu_2}^3} + \frac{b{\mu_1}^2}{{\mu_2}^2} - \frac{c\mu_1}{\mu_2} + d \qquad \text{Eq. 2.62}$$

Using this variable substitution, any Vandermonde effect is avoided and the polynomial solution is effective.
Almost all program compiler have some built-in routines that automatically applies this substitution and search for the best fitting function, being able to provide all parameters needed to evaluate the curvature (firsts and seconds derivatives). in MatLab© this processes can be done with the functions:

- **polyfit**, which calculates the coefficients for a polynomial **p(x)** of degree **n** that is a best fit for the data in **y**. The coefficients in **p** are in descending powers, and the length of **p** is **n+1;**
- **polyval**, which evaluates at x the value of a polynomial of degree **n**. The input argument **p** is a vector of length **n+1** whose elements are the coefficients in descending powers of the polynomial to be evaluated;
- **Polyder** that calculates the derivative of polynomials, polynomial products, and polynomial quotients.

With the programming software MatLab© (after creating owns routines and databases, see [5]) the procedure for all points needed to be investigated were used.
Looking to this methodology from different points of view, the polynomial solution is actually convenient regarding many different issues, like the following.

The polynomial fitting works separately regarding horizontal and vertical alignments and ti generates two different fitting environments: (x, y) coordinates in the horizontal reference and stations/elevations on the vertical side.
This separation can lead to some advantages connected to road engineering traditions: road engineers and practitoners have usually investigated horizontal and vertical alignments in separate ways, so that it is always guaranteed that curvatures on both directions are easily understandable. This condition is not actually guaranteed by, e.g., the 3D method, since the curvature obtained in the space is not exactly the same measured separately on both horizontal and vertical plane. The 3D methodology gives back the *real* curvature in space that the driver feels along the road, but it is not the *design* one.

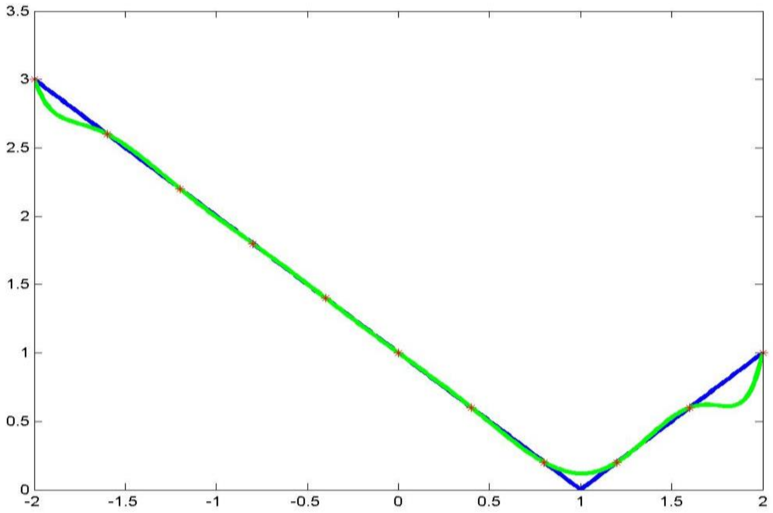


Figure 2.7 - Example of polynomial fitting on tangents

It allows a wide range exploration, since it considers not only three successive epochs but also a variable number of epochs. The accurate set of this parameter is a key factor to avoid noises or imprecisions due to positioning errors, or to be able to set the desired scale analysis.

The polynomial fitting is effective on tangents as well as on curves: differently from circular curves fitting, polynomial functions are well suitable on tangents too, without incurring on machine errors and limits. Tangents are critical sections when they are fitted with circumferences, since they require almost infinite radius and they are really close to machine calculating limits, giving back errors and bad estimations [Figure 2.7].

It is immune to Gibbs phenomena: several fitting methods are affected by the so-called "Gibbs phenomenon". It can eventually occur when it is required that the fitting function passes through surveyed points. In these cases, the function may have a strong irregular behavior, generating big periods and oscillations between points even if the real alignment is completely straight.
That is mainly due to possible local positioning errors that suddenly force the function to bend and induce this "bending" in a big influence area [Figure 2.8]

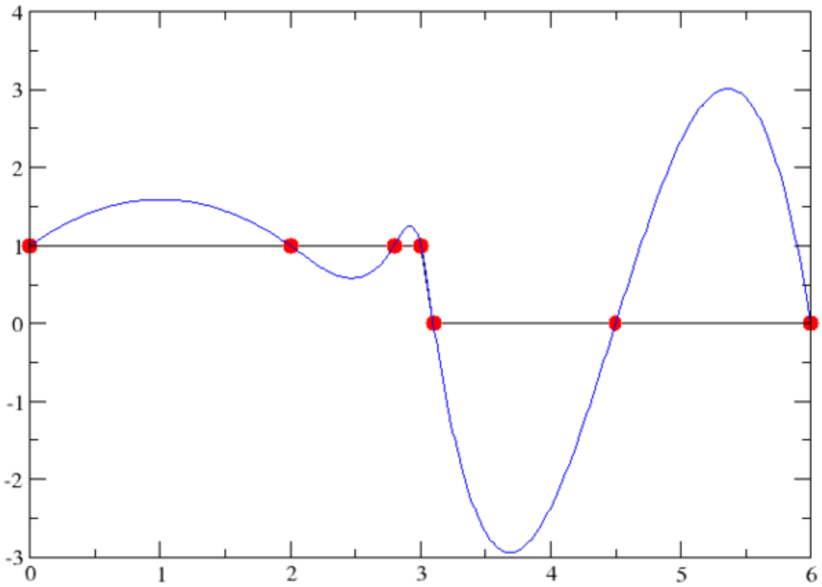


Figure 2.8 - Gibbs phenomena with a spline fitting

Local instability is only possibile at the edges of the area: as also visible in Figure 2.7 when a polynomial fitting is performed it is possible to see some instability at the edge of the investigated area. In this specific applications these behavior does not represent a problem because the curvature extracted from the polynomial is assigned to the central point of the fitting region, so that each point will always be in the center of this area and will not be affected by any instability.

Thanks to all thiese advantages, it has been decided to adopt this technique to evaluate curvatures on the analyzed case study (Marinelli, 2011). Due to the nature of this fitting method, every time that a fitting procedure is performed it is necessary to ideally transform the East coordinates into a set of independent variables (**x**) and the North coordinates into a set of dependent variables (**y**). This means that the polynomial fitting will be the math function that will correlate East to North. Unlikely, this conversion to a function has to to respect all mathematical conditions that are valid for a polynomial function, and the most restrictive one in this case is the function *injectivity* [Figure 2.9].

In mathematics, a function is injective (also called one-to-one function) when it preserves distinctness: it never maps distinct elements of its domain to the same element of its codomain. In other words, every element of the function's codomain is the image of at most one element of its domain. The term one-to-one function must not be confused with one-to-one correspondence (aka surjective injection or bijective function), which uniquely maps all elements in both domain and codomain to each other.

To easily understand the meaning of this problem it's useful to have a look at Figure 2.10, where some sample math functions are shown: the first one is a non-injective spiral, where some points of the curve which have the same abscissa are present, even with different y-values. In other terms, it is not possible to assign to each single point of the function *image* a unique value in its *domain*; instead, in the second function, the injectivity is guaranteed. In Figure 2.10(b) it is shown a real case of road survey where the geometry of the axis does not satisfy injectivity conditions, avoiding using original coordinates to perform the fitting.

To fix this problem it is suggested to adopt a mobile and adaptive reference frame: basically, for each single fitting window, the reference system is rotated and translated into a new one. Two basic movements characterize this new reference:

- rigid translation of the system, with new origin on the first point of the fitting window [Figure 2.11(b)]; and
- Rotation of the reference system, with x axis exactly aligned with the direction between first and last point of the fitting window [Figure 2.11(c)].

Equations involved in this operation are firstly the change of origin:

$$X = \begin{bmatrix} x_1 - x_1 \\ x_2 - x_1 \\ x_3 - x_1 \\ \vdots \\ x_{n-1} - x_1 \\ x_n - x_1 \end{bmatrix} \qquad Y = \begin{bmatrix} y_1 - y_1 \\ y_2 - y_1 \\ y_3 - y_1 \\ \vdots \\ y_{n-1} - y_1 \\ y_n - y_1 \end{bmatrix} \qquad \text{Eq. 2.63}$$

where **X** and **Y** are vectors containing respectively **x** and **y** coordinates of all points in the fitting window and **x_1** and **y_1** are the coordinates of the first point in the fitting window. The second step is the system rotation, with the following:

$$\alpha_{rot} = atan\left(\frac{y_n - y_1}{x_n - x_1}\right) \qquad \text{Eq. 2.64}$$

$$R = \begin{bmatrix} \cos\alpha_{rot} & \sin\alpha_{rot} \\ -\sin\alpha_{rot} & \cos\alpha_{rot} \end{bmatrix} \qquad \text{Eq. 2.65}$$

$$[X_{rot} \quad Y_{rot}] = R \cdot \begin{bmatrix} X^T \\ Y^T \end{bmatrix} \qquad \text{Eq. 2.66}$$

Adopting this technique, it is quite likely that any non-injective situation can be solved and the polynomial is correctly fitted.

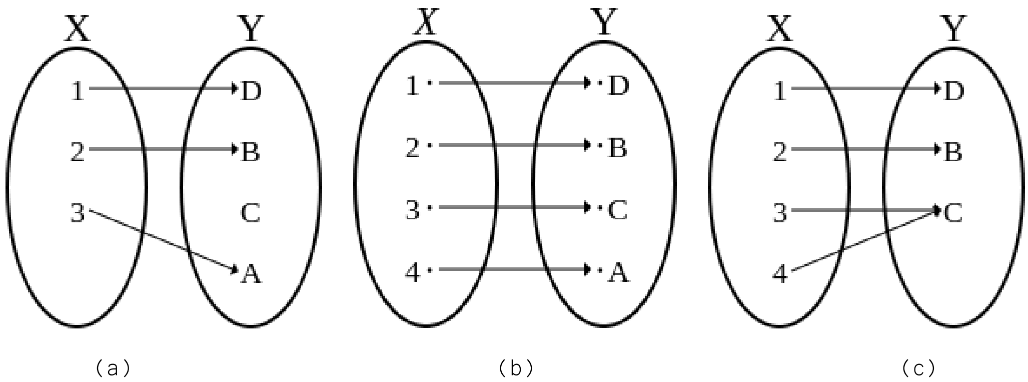


Figure 2.9 - Functions characteristics with Eulero-Venn representation
(a) An injective non-surjective function (b) An injective surjective function (c) A non-injective surjective function

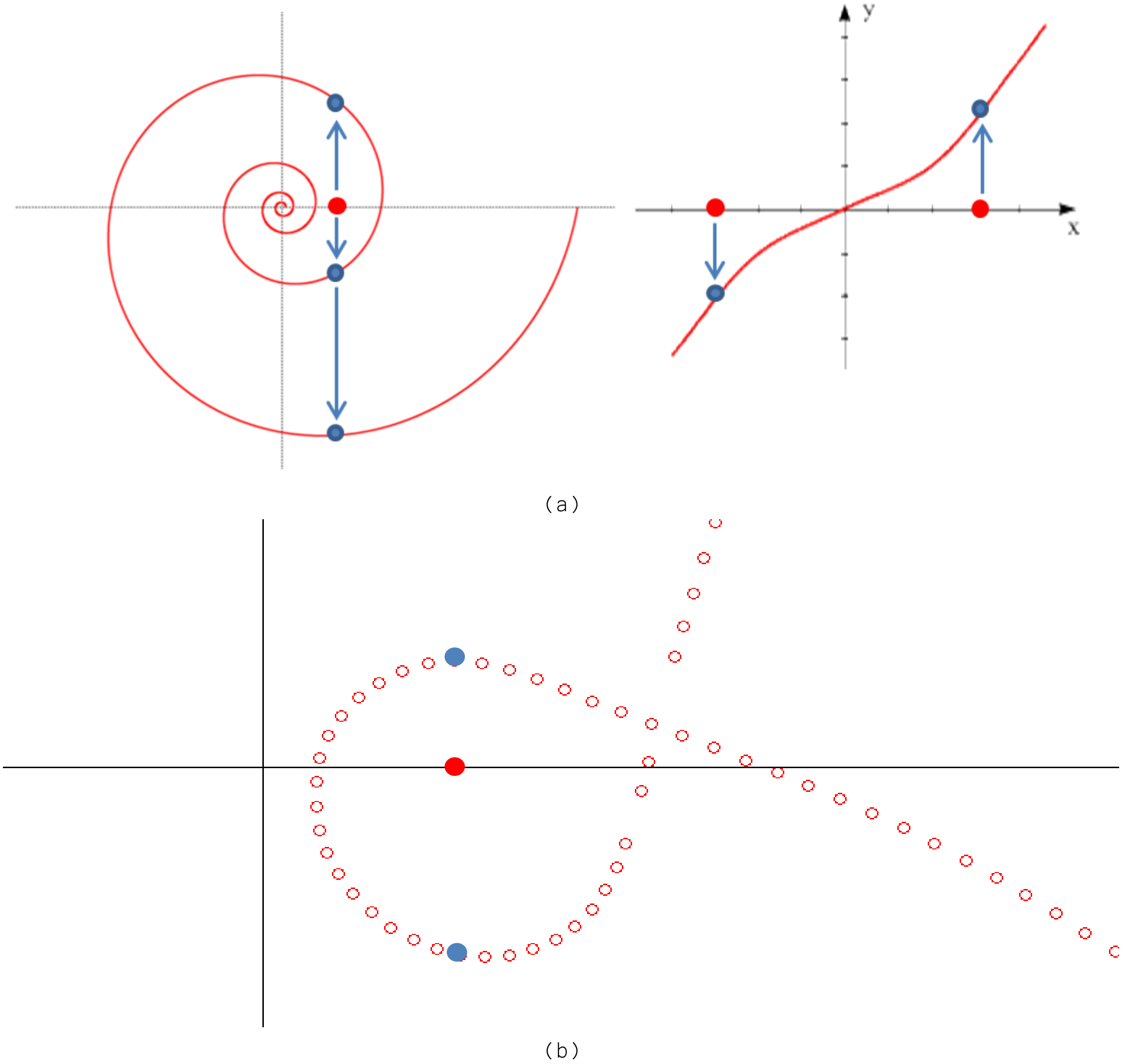


Figure 2.10 – Injectivity samples and practical situations:
(a) A spiral (not injective) and a generic injective function (b) Non-Injective situation inside a motorway interchange

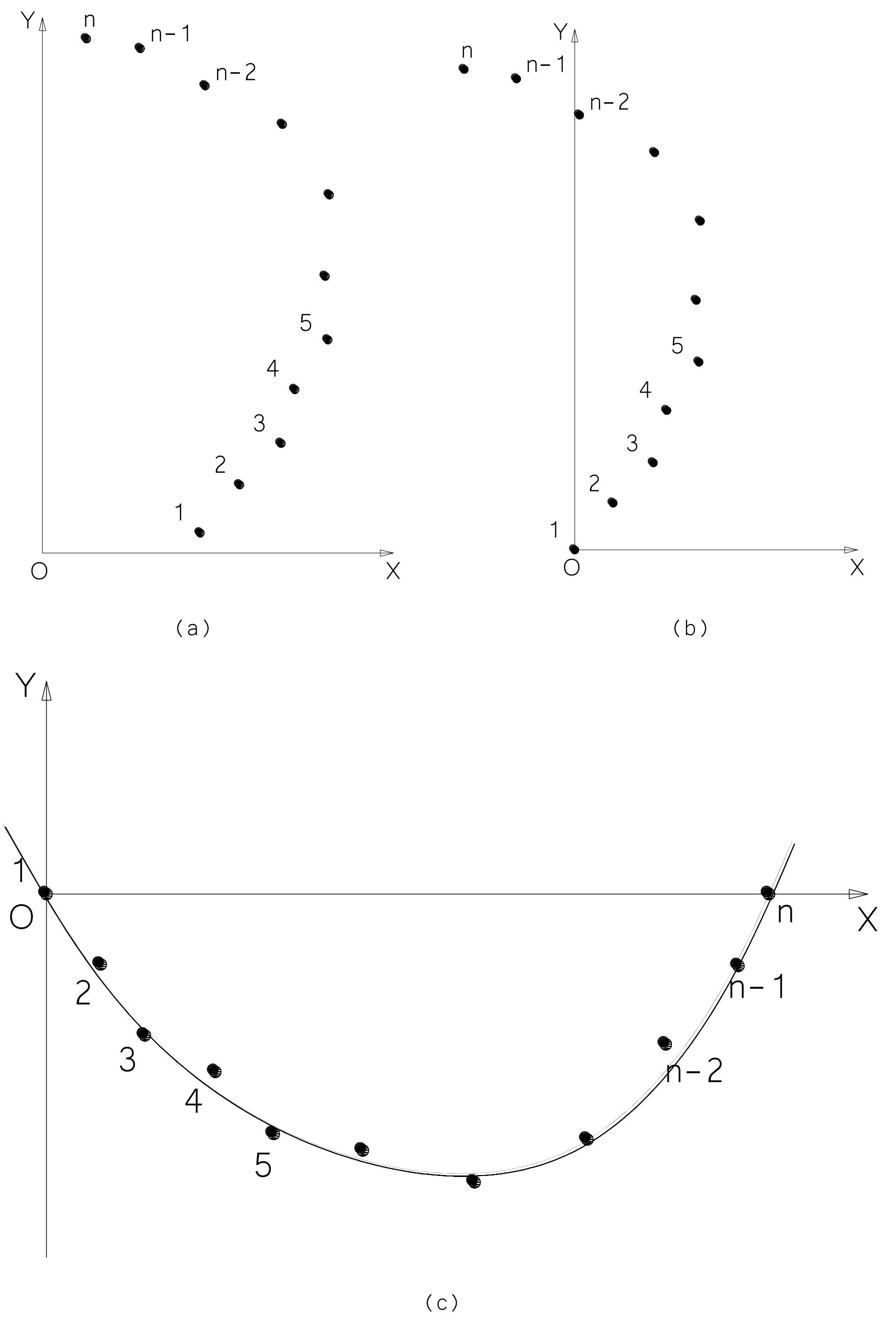


Figure 2.11 - Local reference system change: an example
(a) Starting injective situation (b) Change of system origin (c) Rotation of reference system

Assumed that polynomial functions are the best one to evaluate curvatures, it is necessary to establish which parameters are involved inside the fitting process.

There are two parameters: **fitting window width** and **polynomial function degree.** The fitting window is relevant in order to define how much focused the analysis is desired to be and, conversely, how many points is suggested to take into account in order to eliminate some local mistakes and positioning errors.

Polynomial degree is also supposed to influence the accuracy and fitting capacity of the function: the higher the degree, the more "fitting" and "adapting" is the polynomial. It's necessary to keep in mind that fitting window width and polynomial degree are in some way correlated each other, since it's not possible to apply a fitting algorithm if there aren't enough point to perform it.

This correlation is usually the following:

$$\#points \geq polynomial\ degree + 2 \leftrightarrow \#points \geq \#polynomial\ parameters + 1 \qquad \text{Eq. 2.67}$$

2.4 Looking from a new perspective

Many are the techniques adopted to recognize geometric elements on road surveys: many of them have been introduced in the previous pages [see 1.2.1].

Anyway, local curvatures and deviation angles are the most used and more effective parameters that have to be taken into account to perform a starting analysis of the road survey. As already said, the ability to distinguish each single element of the track is the first needing to perform successively a regression analysis.
Generally, the grouping process starts from the comparison between the calculated data and a minimum threshold: each value (in curvature or deviation angle) that overpasses a specific pre-determined value is considered as belonging to a curved element of the road alignment (curves or clothoids). In fact, fixing this threshold is not easy to do and generally needs for a calibration upon sampled data (Li, Chitturi, Bill, & Noyce, 2012). In fact, each single dataset is determined by a specific acquisition error and needs a different threshold in order to be applied.

So the question could be: *are these thresholds general and absolute or they need to be defined regarding to the specific survey technology and technique implemented case by case?* Is it possible to delineate some guidelines in terms of accuracies, precisions, mathematical issues and similar with respect to the accuracies obtained from the "in field" acquisition?

That is exactly why this research work covers an innovative position: it tries to explore road alignment acquisition, grouping and fitting with a new perspective, instead of working directly on a case study and trying to solve the alignment recognition problem for that specific case.
The aim of this work is giving to practitioners, stakeholders, and researchers some sort of a framework in which they can move regarding road surveys and regressions. Some indications will be provided in terms of accuracies and application capabilities of deviation angles and curvatures, while in next chapters some fitting algorithms will be investigated.
This applied part of the work is contained in part B.

c3 Fitting methods, theories and models

> *"The insight that leads to a new theory about the nature is perhaps more powerful than the artistic creation, because a new physical theory must be not only original but also consistent in terms of mathematics and compatible with the experimental observations.*
> *The world's most elegant theory is useless if it is contradicted by the experiments."*
> ***J. Kakalios – The Physics of Superheroes***

Each time it is necessary to evaluate a certain physical quantity (mass, distanc, volume, pressure, etc.) it must be known that there is no theoretical possibility that the measure obtained from any kind of measurement tool (scale, ruler, barometer, etc.) is exact. In fact, each time a measure is taken some measurement errors occur due to the operator inability, instrument's incorrect calibration, weather conditions, and others. Errors are generally classified upon their origin:

- coarse errors,
- systematic errors, such as
    - instrumental errors,
    - method errors,
    - personal/psychological error, and
- Random errors.

Coarse errors are usually due to carelessness or lack of concentration of the observer, and are the least dangerous, since they occur less frequently and therefore are easily detectable. Instrumental errors, that are part of those systematic, are due to defects in construction of the instrument adopted for the measurement, and all are characterized by having a "direction" (can be all positive or all negative) . Random errors are due to causes of different nature, and are not easily detectable and therefore are quite dangerous.

Due to these errors, it is obvious that it is impossible to determine the exact value of a certain parameter, meaning that "measuring a physical parameter is equal to randomly extract a value from a Gaussian distributed population" (Cina, 2007).

According to this background, when it is required to extract geometrical information from a point set (for example a dynamic survey of a road lane) it is necessary to deal with randomly distributed errors that affect measurements accuracy and generate problems in extracting detailed information. That is why it is necessary to introduce some algorithms that are able to deal with these kinds of errors and give back parameters useful to recognize road alignments, such as radii, length, orientations, scale factors and so on.
There are several techniques that are able to give statistic information of a measurements population, and in this research three of them have been applied: the least squares technique, Landau technique, and Huber one.

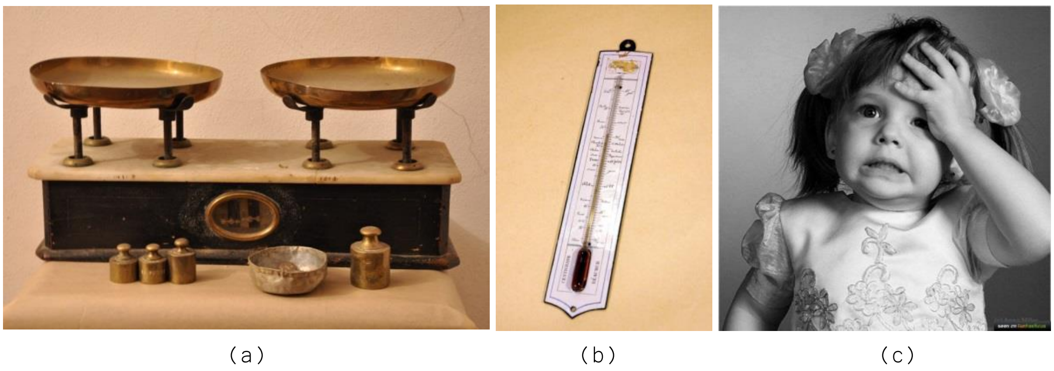


Figure 3.1 – Three different genre of errors
(a) Non-calibrated scale (b) imprecise thermometer (c) Carelessness of the operator (!)

3.1 Least squares

Basically, "least squares" is a specific name applied to the more general "*best likelihood principle*", which is the statistical way to determine and justify why the average of a population of measurements is one of the best possible estimate.

A general equation links together the measure α to the unknown parameters x_i like the following:

$$f(x_1, x_2, \dots, x_r \cdot / \alpha) = 0$$

Eq. 3.1

To simplify the problem, it is assumed that the equation is linear (or it is possible to write it in a linear form), and after several measurements, a system made of **n** equations with **r** parameters is created:

$$Ax + c = \alpha$$

Eq. 3.2

where **c** is a constant known value. Since there is no real possibility to know the exact measurement, it will be possible only to have a good estimate, called here α_0.
Subtracting this quantity to both sides of Eq. 3.2

$$Ax + c - \alpha_0 = \alpha - \alpha_0$$

Eq. 3.3

$$Ax - l_0 = v$$

Eq. 3.4

where

$$l_0 = (\alpha_0 - c)$$

Eq. 3.5

$$v = \alpha - \alpha_0$$

Eq. 3.6

assumed l_0 as the "known element" and **v** as "theoretical errors" vectors. Explaining all terms:

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1r} \\ a_{21} & a_{22} & \cdots & a_{2r} \\ a_{31} & a_{32} & \cdots & a_{3r} \\ \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nr} \end{bmatrix} \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \cdots \\ x_r \end{bmatrix} \quad l_0 = \begin{bmatrix} l_{01} \\ l_{02} \\ l_{03} \\ \cdots \\ l_{0n} \end{bmatrix} \quad v = \begin{bmatrix} v_1 \\ v_1 \\ v_2 \\ \cdots \\ v_r \end{bmatrix}$$

Eq. 3.7

Matrix **A** is called "design matrix" while **x** is the "unknown parameters matrix". Since different operators may take different measurement and, generally, with different accuracies, it's a good practice to introduce a weight, that gives different magnitudes referring to different accuracies, with the following:

$$PAx - Pl_0 = Pv$$

Eq. 3.8

where **P** is the "weight matrix", generally diagonal when the measurement are not correlated.

$$P = \begin{bmatrix} p_1 & 0 & \cdots & 0 \\ 0 & p_1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & p_n \end{bmatrix}$$

Eq. 3.9

Actually the system [Eq. 3.7] is not solvable, because there are **n+r** unknown parameters with **n** equations. Therefore, it is necessary to adopt a statistical methodology, called *least squares*. The target is to find the combination of parameters that minimize the errors from the theoretical value:

$$\sum_i p_i(l_i - l_{0i})^2 = \sum_i p_i v_i^2 \qquad \text{Eq. 3.10}$$

so that the solution will be the closest to the exact value. The problem, now, is to find the minimum of Eq. 3.10, and it is generally solvable finding the zeros on its first derivative; since are available **r** variables, it will be necessary to find **r** zeros of **r** different derivatives:

$$\begin{cases} \dfrac{\partial \sum_{i=1}^n p_i v_i^2}{\partial x_1} = 0 \\ \dfrac{\partial \sum_{i=1}^n p_i v_i^2}{\partial x_2} = 0 \\ \dots \\ \dfrac{\partial \sum_{i=1}^n p_i v_i^2}{\partial x_r} = 0 \end{cases} \qquad \text{Eq. 3.11}$$

Since it is demonstrable that the second derivative of each row of the system is a positive constant (assuming as convenience that r=1):

$$\frac{d\sum_{i=1}^n p_i \cdot v_i^2}{d\widehat{m}} = -2 \cdot \sum_{i=1}^n p_i(x_i - \widehat{m}) \qquad \frac{d^2\sum_{i=1}^n p_i \cdot v_i^2}{d\widehat{m}^2} = 2 \cdot \sum_{i=1}^n p_i \qquad \text{Eq. 3.12}$$

It is possible to surely state that the solution of Eq. 3.11 is actually the desired minimum. That system is called "*normal system*" and can be noted in matrix formulation:

$$N\hat{x} = T_N \qquad \text{Eq. 3.13}$$

$$\hat{x} = N^{-1}T_N \qquad \text{Eq. 3.14}$$

Where **N** is a squared, symmetrical and r-order matrix called "*normal matrix*", **T**$_n$ is the "*known elements normal vector*" with dimension **1xr.**

Adopting this methodology it's necessary, each time there is an estimation problem, to write the errors square summation, than derive it and only after this process is possible to apply an automatic solving procedure for the problem. This way it is not applicable, especially in cases where many measurements are done and many parameters are needed to perform the estimated. Luckily, it is possible to get the solving equation without this process, but only knowing the *design matrix* and the *known elements vector*. In fact, between matrix **A**, **l**, **N** and **T**$_n$ there are some relationships:

$$N = A^T \cdot A \qquad \text{Eq. 3.15}$$

$$T_N = A^T \cdot l_0 \qquad \text{Eq. 3.16}$$

or, more generally with weights:

$$N = A^T PA \qquad \text{Eq. 3.17}$$

$$T_N = A^T P l_0 \qquad \text{Eq. 3.18}$$

Applying some elaborations:

$$\begin{gathered} N\hat{x} = T_N \\ A^T PA\hat{x} = A^T PAl_0 \\ T_N = A^T \cdot \end{gathered} \qquad \text{Eq. 3.19}$$

so that:

$$\hat{x} = (A^T P A)^{-1} A^T P A l_0$$ Eq. 3.20

The fitting problem does not end with parameters estimation, since it is also necessary to estimate their variances. In this case, since the indirect measurement belong to a normal distribution with **r** dimensions, it is necessary to estimate the variance/covariance matrix. It will be a **r**-order square matrix, symmetric with respect to the main diagonal, where all indirect measurements variances are included, while other elements will be covariances.
Thanks to Eq. 3.20, the variance propagation law is appliable, going from the measurement variance/covariance matrix (C_{LL}) to the one on estimated parameters (C_{xx}). Since

$$C_{LL} = \frac{\sigma_0^2}{P}$$ Eq. 3.21

and remembering the following linear algebra theorem

$$(AB)^T = B^T A^T$$ Eq. 3.22

then

$$C_{\hat{x}\hat{x}} = (N^{-1}A^T P)C_{LL}(N^{-1}A^T P)^T = (N^{-1}A^T P)C_{LL}(A^T P)^T N^{-1T}$$ Eq. 3.23

Remembering normal and weight matrixes property **N=N**^{T} and **P=P**^{T}

$$C_{\hat{x}\hat{x}} = (N^{-1}A^T P)C_{LL}PAN^{-1} = (N^{-1}A^T P)\frac{\sigma_0^2}{P}PAN^{-1} = \sigma_0^2(N^{-1}A^T PAN^{-1})$$ Eq. 3.24

Substituting with **N** thanks to Eq. 3.17

$$C_{\hat{x}\hat{x}} = \sigma_0^2(N^{-1}A^T PAN^{-1}) = \sigma_0^2(N^{-1}NN^{-1}) = \sigma_0^2 N^{-1}$$

$$C_{\hat{x}\hat{x}} = \sigma_0^2 N^{-1} = \begin{bmatrix} \sigma_{x_1}^2 & \sigma_{x_1x_2} & \cdots & \sigma_{x_1x_r} \\ \sigma_{x_2x_1} & \sigma_{x_2}^2 & \cdots & \sigma_{x_2x_r} \\ \cdots & \cdots & \cdots & \cdots \\ \sigma_{x_rx_1} & \sigma_{x_rx_2} & \cdots & \sigma_{x_r}^2 \end{bmatrix}$$ Eq. 3.25

3.1.1 Road alignment fitting equations

All the equations explained in 3.1 are extremely general and are suitable for many different applications cases and fields. Obviously, to be applied to road alignment reconstruction it has been necessary to evaluate all the parameters for road geometrics, and in particular focusing on circles fitting, tangents, and clothoids.
Generally, the database on which all the elaborations are usually done is a points list, characterized by 4 main information fields:

- East coordinate
- North coordinate
- Elevation
- Acquisition time

In fact, each point is precisely defined in time and space: the first one is the GNSS exact acquisition time, while the second one is the position given in a certain reference and coordinate systems. Usually, the main adopted coordinate system is the UTM-WGS84, while sensors often returns information using a geographical coordinate system, such as latitudes and longitudes. This means that is usually necessary a coordinate conversion from one system to the other.

3.1.1.1 From geographic to cartographic coordinates

To solve this conversion problem the main adopted methodology is the one set up by Hirvonen (Inghilleri, 1974), which are not the only one available in literature but they are really effective because they converge and they are able to reach the millimeter accuracy. With this conversion, there is no change in reference system, but only a change in coordinate system, from a geographic one to a cartographic one.
These equations are directly dependent from the Gauss representation [Figure 3.2]; the horizontal coordinate reference system has the **y** axis on the single spindle central meridian and the **x** axis on the equator.

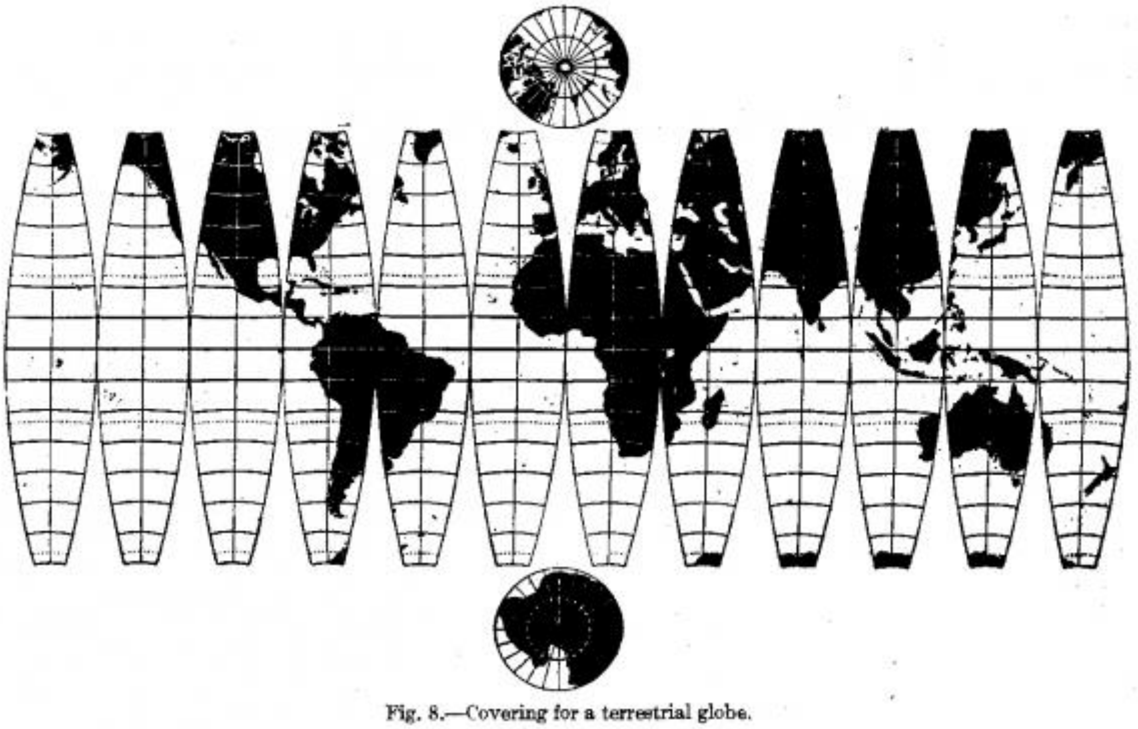


Fig. 8.—Covering for a terrestrial globe.

Figure 3.2 - Gauss representation

Hirvonen formulations are always referred to the spindle central meridian, so it will be always necessary to depurate from fake origins.
Direct formulations are:

$$x = d \cdot arcsinh \frac{cosz{\cdot}tan\lambda'}{v_1} \qquad \text{Eq. 3.26}$$

$$y = a(A_1z - A_2sin2z + A_3sin4z - A_4sin6z) \qquad \text{Eq. 3.27}$$

Going into details of each element:

$$\text{arcsinh}(t) = log\left(t + \sqrt{1 + t^2}\right) \qquad \text{Eq. 3.28}$$

$$\varepsilon^2 = \frac{a^2 - c^2}{c^2} = \frac{e^2}{1 - e^2} \qquad d = \frac{a^2}{c} \qquad \text{Eq. 3.29}$$

$$v = \sqrt{1 + \varepsilon^2 cos^2\varphi} \qquad v_1 = \sqrt{1 + \varepsilon^2 cos^2 z} \qquad \text{Eq. 3.30}$$

$$\lambda' = (\lambda - \lambda_0) \qquad \text{Eq. 3.31}$$

$$z = arctan \frac{tan\varphi}{cos(\lambda' v)} \qquad \text{Eq. 3.32}$$

$$A_1 = \left(1 - \frac{e^2}{4} - \frac{3e^4}{64} - \frac{5e^6}{256}\right) \quad \text{Eq. 3.33}$$

$$A_2 = \left(\frac{3e^2}{8} + \frac{3e^4}{32} + \frac{45e^6}{1024}\right) \quad \text{Eq. 3.34}$$

$$A_3 = \left(1 - \frac{15e^4}{256} + \frac{45e^6}{1024}\right) \quad \text{Eq. 3.35}$$

$$A_4 = \left(\frac{35e^6}{3072}\right) \quad \text{Eq. 3.36}$$

where:

- **a** = equatorial semi axis
- **c** = polar semi axis
- **λ** = spindle central meridian longitude
- **e** = ellipsoid eccentricity

Cartographic coordinates East and North are calculated from **x** and **y**, applying the contraction factor **m_c** and fake origins **x_o** and **y_o**:

$$East = m_c x + x_0 \qquad North = m_c y + y_0 \quad \text{Eq. 3.37}$$

In the following table, there is a summary of the main parameters needed (valid for WGS84).

	WGS84	
A_1	0.998324298453	
A_2	0.002514607060	
A_3	0.000002639047	
A_4	0.000000003418	
a	6378137 m	
e^2	0.006694379990	
Δλ	6°	
$λ_0$	East GW= 9°	West GW= 15°
x_o	500 km	
y_o	North= 0 km	South= 10000 km
m_c	0.9996	

Table 3.1 – International geodetic and cartographic conventions

3.1.1.2 Least squares applied in road geometrics

In order to apply least squares techniques to road alignment elements it's necessary to build, case by case, the design matrix **A**, the weight matrix **P** and the known terms vector **l_o**. In fact, adopting [Eq. 3.20] it is enough to know these three matrixes and it is possible to estimate the desired parameters. The general procedure to extract these matrixes for each element is to start from the equation that links together the researched parameters and field measurements.

Circular curves, transition elements (clothoids) and tangents generally compose road alignment. Each one has different parameters that have to be estimated, but not all of them can be directly fitted from field data. Circular curves and tangents, in fact, have a finite equation that can be easily implemented in a fitting least squares based algorithm, but not the same can be stated about clothoids, that are generally extracted in a second step. That is why it is generally preferred to start estimating curves and tangents, and then extract clothoids parameters.

The main parameter about curves is the radii and then is possible to add all other parameters derived from it. Since during the survey the curve is represented by a series of points, it is necessary to start from the equation that links the radius of a circumference to the points that belongs to it.

Generally, the circumference with center (α,β) and radius **R**, is the geometrical "space" with the following property:

$$(x-\alpha)^2+(y-\beta)^2=R^2 \qquad \text{Eq. 3.38}$$

namely the grouping of all points (and only those ones) which distance from the origin is exactly R.

This general equation is usually matched with a more formal one; superimposing that:

$$-\frac{a}{2}=\alpha \qquad -\frac{b}{2}=\beta \qquad c=\alpha^2+\beta^2-R^2 \qquad \text{Eq. 3.39}$$

it's possible to get a second useful equation:

$$x^2+y^2+ax+by+c=0 \qquad \text{Eq. 3.40}$$

Eq. 3.40 represents actually the equation of a general point inside a circumference, containing three unknown parameters that define the circumference in the plane {x,y}: this means that to get all the parameters of the curve it's necessary to estimate **a**, **b** and **c** and then, using Eq. 3.39, calculate **R**. Surveys usually gives back point coordinates, so that the starting equation in order to obtain the design matrix **A** deals with point coordinates. In fact, generalizing this equation to all points that are part of a road circular curve, the following system is obtained:

$$\begin{cases} x_1^2+y_1^2+ax_1+by_1+c=0 \\ x_2^2+y_2^2+ax_2+by_2+c=0 \\ x_3^2+y_2^2+ax_3+by_3+c=0 \\ \vdots \\ x_n^2+y_n^2+ax_n+by_n+c=0 \end{cases} \qquad \text{Eq. 3.41}$$

With the insulation of all known parameters in a side of the equation:

$$\begin{cases} ax_1+by_1+c=-x_1^2-y_1^2 \\ ax_2+by_2+c=-x_1^2-y_1^2 \\ ax_3+by_3+c=-x_1^2-y_1^2 \\ \vdots \\ ax_n+by_n+c=-x_1^2-y_1^2 \end{cases} \qquad \text{Eq. 3.42}$$

There is no necessity to linearize the equation because it is already linear, so it is easy to move to matrix form:

$$A=\begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \\ \cdots & \cdots & \cdots \\ x_n & y_n & 1 \end{bmatrix} \qquad x=\begin{bmatrix} a \\ b \\ c \end{bmatrix} \qquad l_0=\begin{bmatrix} -x_1^2-y_1^2 \\ -x_2^2-y_2^2 \\ -x_3^2-y_3^2 \\ \cdots \\ -x_n^2-y_n^2 \end{bmatrix} \qquad \text{Eq. 3.43}$$

The same process can be done with tangents, starting from the fundamental equation:

$$y=mx+q \qquad \text{Eq. 3.44}$$

Generalizing with **n** points:

$$\begin{cases} y_1 = mx_1 + q \\ y_2 = mx_2 + q \\ y_3 = mx_3 + q \\ \vdots \\ y_n = mx_n + q \end{cases} \qquad \text{Eq. 3.45}$$

and, into to the matrix form:

$$A = \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ x_3 & 1 \\ \cdots & \cdots \\ x_n & 1 \end{bmatrix} \quad x = \begin{bmatrix} m \\ q \end{bmatrix} \quad l_0 = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \cdots \\ y_n \end{bmatrix} \qquad \text{Eq. 3.46}$$

In both cases, it is necessary to implement the weight matrix **P**, but inserting this process inside an automatic algorithm is definitely simple, since all matrixes and vectors are very and only dependent from point coordinates.

That procedure leads to the estimation of circular curves and tangents. The third element that generally has to be estimated too is the transition curve, but actually, for this element, it has not performed a proper fitting, but a mathematical calculation based upon curves and tangents parameters, pre-estimated through the processes already illustrated. In order to estimate the clothoid **A** parameter it's necessary to know two other parameters: **R** and **ΔR**. The first one is simply the radius of the adjacent circular curve, while ΔR is the offset between the circular curve and the tangent, as shown in Figure 3.3.
The equation of **ΔR** is not a finite equation and has the following general form:

$$\Delta R = \frac{A^4}{24R^3} \cdot \left[\sum_{i=1}^{\infty} (-1)^{i+1} \cdot \frac{6 \cdot \tau_f^{(21-2)}}{(21)! \cdot (4i-1)} \right] \qquad \text{Eq. 3.47}$$

where, generally, it's also allowed to stop the series development at the first term.
It is also possible to determine **ΔR** in function of **L**, because it's enough the following

$$R \cdot L = A^2 \qquad \text{Eq. 3.48}$$

and so

$$\Delta R = \frac{A^4}{24R^3} = \frac{L^2}{24R} \qquad \text{Eq. 3.49}$$

Now is immediate to extract **A,** knowing **R** and **ΔR**:

$$A = \sqrt[4]{\Delta R \cdot 24R^3} \qquad \text{Eq. 3.50}$$

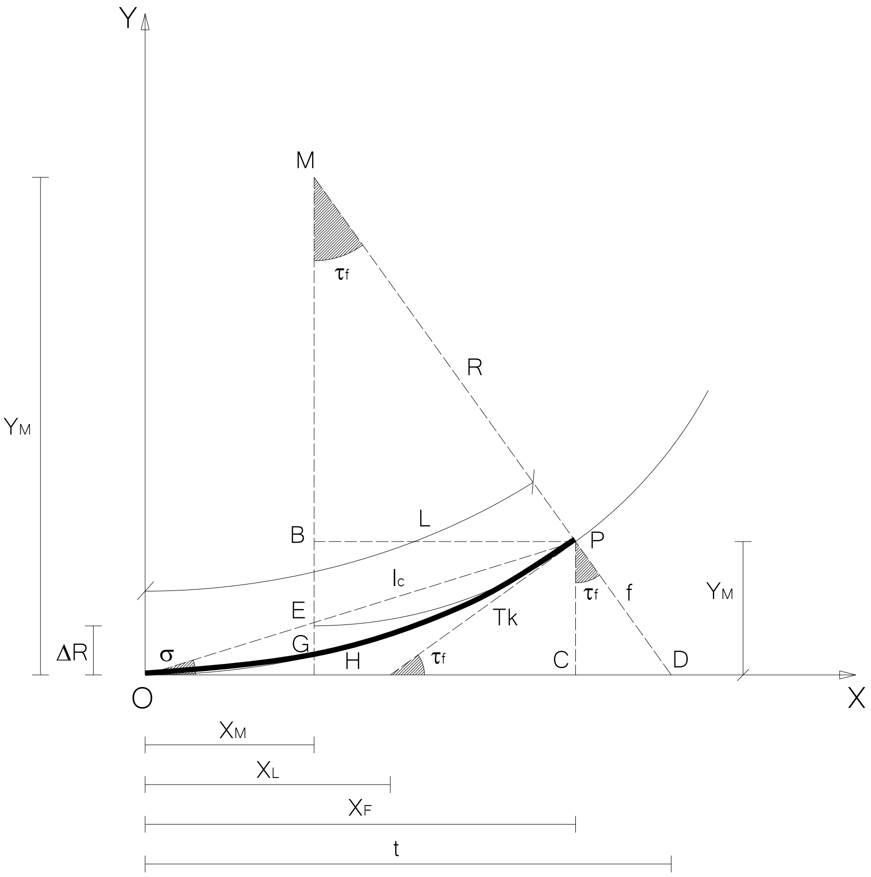


Figure 3.3 – Clothoid general scheme

When all parameters have been estimated or back calculated, variances can be evaluated thanks to Eq. 3.25. Curves and tangents covariances will be characterized by different matrixes, based on different parameters:

- Curves = estimated parameters **a**, **b**, **c** (see Eq. 3.40)

$$C_{\hat{x}\hat{x}} = \begin{bmatrix} \sigma_a^2 & \sigma_{ab} & \sigma_{ac} \\ \sigma_{ba} & \sigma_b^2 & \sigma_{bc} \\ \sigma_{ca} & \sigma_{cb} & \sigma_c^2 \end{bmatrix} \qquad \text{Eq. 3.51}$$

- Tangents = estimated parameters **m**, **n** (see Eq. 3.44)

$$C_{\hat{x}\hat{x}} = \begin{bmatrix} \sigma_m^2 & \sigma_{mn} \\ \sigma_{nm} & \sigma_n^2 \end{bmatrix} \qquad \text{Eq. 3.52}$$

3.2 Robust fitting: the Huber estimator

When least squares regression is performed using *n* observations certain idealized assumptions are made about the vector of errors ε, namely, that is distributed $N(0, I\sigma^2)$. In practice, departures from these assumptions occur. If the departures are serious, there is the hope to spot them in the behavior of the residuals and so to be led to make suitable adjustments to the model and/or the variables' metrics. As an example, it could be possible to transform the response variable, one or more of the predictors, or adjust the model by adding higher-order terms. For many sets of data, the departures, if they exist at all, are not serious enough for corrective actions, and the analysis proceeds in the usual way.

If the analysis seems to point to the errors having a non-normal distribution, it could be considered a robust regression method, particularly in cases where the error distribution is heavier-tailed then the normal, that is, has more probability in the tails than the normal. Such heavier-tailed distribution is likely to generate more "large" errors than the normal. A least squares analysis weights each observation equally in getting parameter estimates. Robust methods enable the observations to be weighted unequally. Essentially, observations that produce large residuals are down-weighted by a robust estimation method. A number of methods are available. In general, robust regression methods require much more computing than least squares, and require some assumptions to be made about the down-weighting procedure to be employed.

From literature, the various methods can be divided into the following three groups

- methods for problems with outliers in the y-direction, where the most common procedures for solving these types of problem are based on the Huber estimator;
- methods applied where there is a moderate percentage of outliers in the coordinate space, which is also known as the leverage point; the estimators, also known as bounded influence estimators, which are employed to solve this type of problem are the Mallows and Schweppe type estimators;
- Methods used when the frequency of outliers in both the y and x directions is high; in this case high breakdown estimators are used.

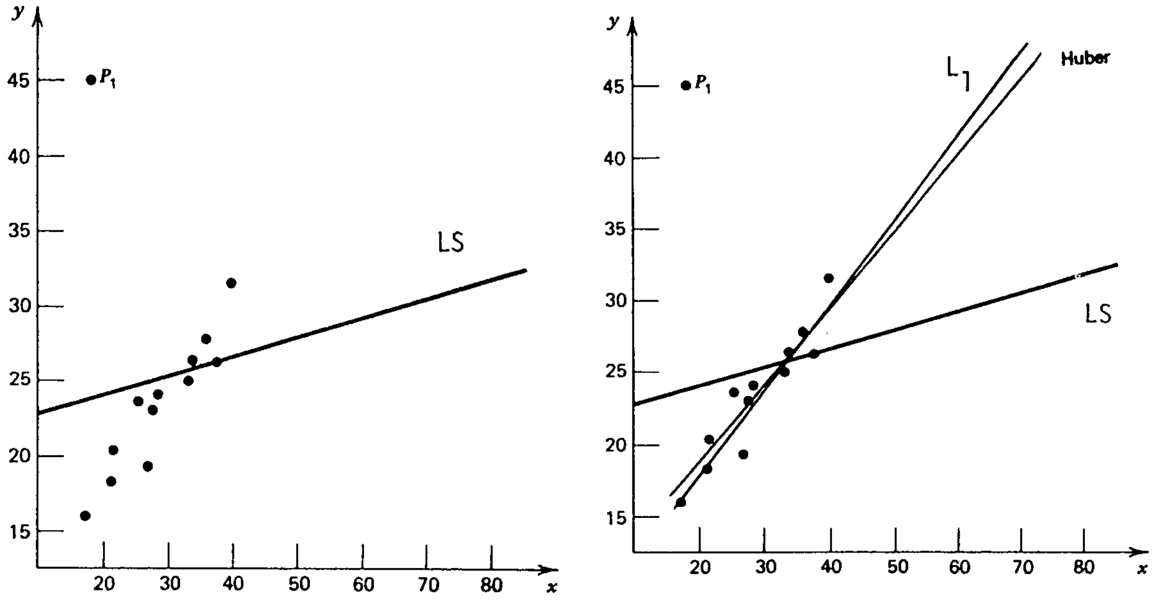


Figure 3.4 - Fitting process with Least Squares method and Robust regressions

3.2.1 Least absolute deviations regression (L₁ regression)

When it is possible to make a specific (non-normal) assumption about the errors, maximum likelihood methods is directly appliable. The assumption of a double exponential distribution

$$\frac{1}{2\sigma}e^{\frac{-|\varepsilon|}{\sigma}} \qquad (-\infty \leq \varepsilon \leq \infty) \qquad \text{Eq. 3.53}$$

leads to minimizing

$$\sum_{i=1}^{N}|\varepsilon_i| \qquad \text{Eq. 3.54}$$

the sum of absolute errors. This is also called L_1 regression (or L_1-norm regression), where the subscript 1 refers to the power used in the function to be minimized. This method gives less weight to larger errors than the least squares method, where the error distribution is

$$\frac{1}{2\pi\sigma}e^{\frac{-\varepsilon^2}{2\sigma^2}} \qquad \text{Eq. 3.55}$$

and the function minimized is the sum of squares of errors, namely,

$$\sum_{i=1}^{N}{\varepsilon_i}^2 \qquad \text{Eq. 3.56}$$

Thus, L_1 regression is more robust against large errors than least squares; least squares are sometimes called L_2 regression (or L_2-norm regression). There exist also L_p regression methods that minimize

$$\sum_{i=1}^{N}|\varepsilon_i|^p \qquad \text{Eq. 3.57}$$

It has been suggested by Forsythe (1972), based on a 400-replicates Monte Carlo study, that a value of p=1.5 could be a good general choice. It led to estimate that were no worse than 95% as efficient as least squares when the errors were actually normal. Efficiency was defined as the ration (mean square error for least squares)/(mean square error for power p), and Forsythe's Monte Carlo experiments used p = 1.25, 1.50 and 1.75 on sets of straight line regression data with errors generated from a normal distribution contaminated with another off-center normal to produce a skewed distribution of errors.

3.2.2 M-estimators

M-estimators are "maximum likelihood type" estimators. Suppose the errors are independently distributed and all follow the same distribution, $f(\varepsilon)$. Then the maximum likelihood estimator (MLE) of β is given by $\hat{\beta}$, which minimizes the quantity

$$\prod_{i=1}^{n} f(Y_i - {x_i}'\beta) \qquad \text{Eq. 3.58}$$

Where ${x_i}'$ is the i-th row of X, i=1,2,3,...,n, in the model $Y = X\beta + \varepsilon$. Equivalently, the MLE of β maximizes

$$\sum_{i=1}^{n} lnf(Y_i - x_i{}'\beta) \qquad \text{Eq. 3.59}$$

As it has been shown, this leads to minimizing the sum of squares function

$$\sum_{i=1}^{n} (Y_i - x_i{}'\beta)^2 \qquad \text{Eq. 3.60}$$

in the normal case. In the double exponential case it's minimized

$$\sum_{i=1}^{n} |Y_i - x_i{}'\beta| \qquad \text{Eq. 3.61}$$

This idea can be extended as follows. Suppose $p(u)$ is a defined function of u and suppose s is an estimate of scale (not necessarily the usual least squares estimate). A robust estimator is defined as one that minimizes

$$\sum_{i=1}^{n} p\left(\frac{e_i}{s}\right) = \sum_{i=1}^{n} p\left(\frac{Y_i - x_i{}'\beta}{s}\right) \qquad \text{Eq. 3.62}$$

It is clear that if $\mathrm{p(u) = u^2}$, the criterion minimized is the same as Eq. 3.60, while if $\mathrm{p(u) = |u|}$, it is obtained Eq. 3.61. So in these specific cases, the form of p and the underlying distribution are specifically related. In fact, the *knowledge* of an appropriate distribution for the error could explain what $p(u)$ to use.

Here, is encountered the first practical difficulty in using robust regression. In general, if the distribution to assume for the errors is not known, it is impossible to be led naturally to a $p(u)$. Essentially, in literature there are a number of suggestions for $p(u)$ that have worked out well in specific studies. It is not clear which should be used for a given set of data. The only thing that can be done is to look at the weighting characteristics produced by $\mathrm{p(u)}$ and choose accordingly, all else being equal.

Table 3.2 shows some of the suggestions given in the literature for $\mathrm{p(u)}$. A choice of specific values for the constants a, b, and c needs to be made; values shown are suggested in literature. In Eq. 3.62, the role of u is played by the scaled residual $(Y_i - x_i{}'\beta)/s$.
The least squares method gives the highest weight of 1 in Eq. 3.62 to larger scaled residuals, and the purpose of the various $\mathrm{p(u)}$ functions is to (comparatively) down-weight larger scaled residuals in various ways, compared with least squares. Let us examine how the estimation procedure is carried out.

3.2.3 The M-estimation procedure

It is needed to minimize Eq. 3.62 with respect to parameters $\beta_j, j = 0,1,2, \dots, k,$ say. Differentiation of Eq. 3.62 with respect to each parameter leads to $\mathrm{p = k + 1}$ equation of form

$$\sum_{i=1}^{n} x_{ij}\psi\left(\frac{Y_i - x_i{}'\beta}{s}\right) = 0 \qquad j = 0,1,2, \dots, k \qquad \text{Eq. 3.63}$$

where $\psi(u)$ is the partial derivative $\frac{\partial p}{\partial u}$ and x_{ij} is the j-th entry of $x_i{}' = (1, x_{i1}, x_{i2}, \dots, x_{in})$. These equations do not have an explicit solution in general, and iterative numerical solution is necessary. It is possible to follow Beaton & Tukey (1974) and define weights

$$w_{i\beta} = \frac{\psi\left(\frac{Y_i - x_i'\beta}{s}\right)}{\frac{Y_i - x_i'\beta}{s}} \qquad i = 1,2,3, \dots, n \qquad \text{Eq. 3.64}$$

defining them to be 1 if it happens that $\mathrm{Y_i - x_i'\beta = 0}$ exactly. Then Eq. 3.63 becomes

$$\sum_{i=1}^{n} x_{ij} w_{i\beta} (Y_i - x_i'\beta) = 0 \qquad j = 0,1,2, \dots, k \qquad \text{Eq. 3.65}$$

or

$$\sum_{i=1}^{n} x_{ij} w_{i\beta} x_i'\beta = \sum_{i=1}^{n} x_{ij} w_{i\beta} Y_i \qquad j = 0,1,2, \dots, k \qquad \text{Eq. 3.66}$$

This can be written in matrix language as

$$X'W_\beta X\beta = X'W_\beta Y \qquad \text{Eq. 3.67}$$

where W_β= diagonal $w_{1\beta}, w_{2\beta}, w_{3\beta}, \dots, w_{n\beta}$. These equations are obviously of the form of generalized least squares normal equations. (More accurately, there are *weighted* least squares type because W_β is a diagonal matrix of weights). The difficulty in solving them is that W_β depends on β. This is circumvented in a traditional way. We somehow get an initial estimate $\hat{\beta}_0$ of β (perhaps by using least squares), calculate the value W_0, say, of $\mathrm{W_\beta}$ for that $\hat{\beta}_0$ and solve Eq. 3.71 to get solution $\hat{\beta}_1$. Using $\hat{\beta}_1$ in $\mathrm{W_\beta}$ gets to $\mathrm{W_1}$. Then $\mathrm{W_1}$ is used to get $\hat{\beta}_2$, and so on. Typically, convergence occurs quite quickly but the procedure can also be halted after a selected number of steps, if desired. It is possible to write the iterative solution as

$$\hat{\beta}_{q+1} = \left(X'W_q X\right)^{-1} X'W_q Y \qquad q = 0,1,2, \dots \qquad \text{Eq. 3.68}$$

and the procedure may be stopped when all the estimates change by less than some selected preset amount, say, 0.1% or 0.01%, or after a selected number of steps.

In Eq. 3.64 it's visible that the weights depend on the values of $\frac{\frac{\partial p}{\partial u}}{u}$, where $u = \frac{(Y_i - x_i'\beta)}{s}$. Thus, it is needed to have a look at the function $w(u) = \frac{\frac{\partial p}{\partial u}}{u}$ for all the various choices of $\mathrm{p(u)}$ in Table 3.2.

In this research work, the Huber estimator was adopted (Huber P. J., 1964). In the Matlab® language the statistic toolbox denominated **robustfit** was used at this scope, with a breackpoint that was set equal to **1.345.**

Criterion	$p(u)$	Corresponding ranges of u	$\psi(u) = \frac{\partial p(u)}{\partial u}$	$w(u) = \frac{\psi(u)}{u}$	Suggested parameter or breakpoint values （tuning constants） and reference
（a） Least squares （normal errors）	$\frac{1}{2}u^2$	$-\infty \leq u \leq \infty$	u	1	Not applicable
（b） Huber's, with breakpoint $a > 0$	$\begin{cases} \frac{1}{2}u^2 \\ a\lvert u\rvert - \frac{1}{2}a^2 \end{cases}$	$-a \leq u \leq a$ $u \leq -a \; and \; u \geq a$	u $a \; sign(u)$	1 $\frac{a}{\lvert u\rvert}$	a=2 （Huber P. J., 1964）
（c） Ramsay's with parameter $a > 0$	$\frac{1 - e^{-a\lvert u\rvert}(1 + a\lvert u\rvert)}{a^2}$	$-\infty \leq u \leq \infty$	$ue^{-a\lvert u\rvert}$ $(maximum \; at \; a^{-1})$	$e^{-a\lvert u\rvert}$	a=3 （Ramsay, 1977）
（d） Andrew's wave, with breakpoint $a > 0$	$\begin{cases} a\left[1 - cos\left(\frac{u}{a}\right)\right] \\ 2a \end{cases}$	$-a\pi \leq u \leq a\pi$ $u \leq -a\pi \; and \; u \geq a\pi$	$sin\left(\frac{u}{a}\right)$ 0	$\frac{sin\left(\frac{u}{a}\right)}{\frac{u}{a}}$ 0	a=1.339 （Andrews, et al., 1972）
（e） Tukey's biweight, with breakpoint $a > 0$	$\begin{cases} \frac{1}{2}u^2 - \frac{u^4}{4a^2} \\ \frac{1}{4}a^2 \end{cases}$	$-a \leq u \leq a$ $u \leq -a \; and \; u \geq a$	$u\left(1 - \frac{u^2}{a^2}\right)$ 0	$1 - \frac{u^2}{a^2}$ 0	$5 \leq a \leq 6$ （Beaton & Tukey, 1974）
（f） Hampel's, with breakpoints $a, b, c > 0$	$\begin{cases} \frac{1}{2}u^2 \\ a\lvert u\rvert - \frac{1}{2}a^2 \\ a\frac{\left(c\lvert u\rvert - \frac{1}{2}u^2\right)}{c - b} - \frac{7a^2}{6} \\ a(b + c - a) \end{cases}$	$-a \leq u \leq a$ $b \leq u \; \leq -a \; and \; a \leq u \leq b$ $c \leq u \; \leq -b \; and \; b \leq u \leq c$ $u \leq -c \; and \; u \geq c$	u $a \; sign(u)$ $\frac{[csign(u) - u]a}{c - b}$ 0	1 $\frac{a}{\lvert u\rvert}$ $\frac{\frac{c}{\lvert u\rvert - 1}a}{c - b}$ 0	A=1.7, b=3.4, c=8.5 （Andrews, et al., 1972）
（g） Assume errors follow t_f distribution and apply maximum likelihood		$-\infty \leq u \leq \infty$		$\frac{f + 1}{f + u^2}$	Not applicable As $f \rightarrow \infty$, the t-distribution → normal distribution

Table 3.2 - Some functions that have been suggested for M-estimation

3.3 LANDAU GEOMETRICAL FITTING

Alternative methods could come from other disciplines, like mechanical engineering, where quality control and inspection of mechanical parts often requires the estimation of an arc center and its radius. Even if these specific routines and methods are not properly designed for road engineering purposes, they are suitable for any other applications.

U.M Landau (1987) has suggested an iterative algorithm for estimating the location of circular arc center and its radius. This algorithm was based on the minimization of the error between a set of given points and the estimated arc. During the demonstration, due to the impossibility to exactly solve all the equations needed, Landau suggested an iterative process to solve the system. With a new approach, proposed by Thomas & Chan (1989), it is possible to find an exact solution if the estimated error is carefully redefined: the unsolvable Landau equation can be, instead, exactly solved changing the notion of what the "measure" is for best fitting in a 2-dimensional problem. They proposed that the "measure" is an area, instead of a length.

Given a set of coordinates $(x_1, y_1) \dots (x_i, y_i) \dots (x_N, y_N) \dots$ a circle with center $(\bar{x}, \bar{y})$ and radius R are defined. The error is defined as the difference between the constant area πR^2 and the area of the circle centered at $(\bar{x}, \bar{y})$ and has a radius

$$\sqrt{(x_i - \bar{x})^2 + (y_i - \bar{y})^2} \qquad \text{Eq. 3.69}$$

Summing up the squares of the errors it is obtained

$$e(R, \bar{x}, \bar{y}) = \sum_{i=1}^{N} \{\pi R^2 - \pi[(x_i - \bar{x})^2 + (y_i - \bar{y})^2]\}^2 \qquad \text{Eq. 3.70}$$

or

$$J = \frac{e}{\pi^2} = \sum_{i=1}^{N} \{R^2 - [(x_i - \bar{x})^2 + (y_i - \bar{y})^2]\}^2 \qquad \text{Eq. 3.71}$$

The function $\mathrm{J}(\mathrm{R}, \bar{\mathrm{x}}, \bar{\mathrm{y}})$ should be minimized with respect to $\mathrm{R}, \bar{\mathrm{x}}$ and $\bar{\mathrm{y}}$. Differentiating Eq. 3.71 with respect to R yelds

$$\frac{\partial J}{\partial R} = 2 \sum_{i=1}^{N} \{R^2 - [(x_i - \bar{x})^2 + (y_i - \bar{y})^2]\} \cdot 2R = 0 \qquad \text{Eq. 3.72}$$

or

$$NR^2 = \sum_{i=1}^{N} [(x_i - \bar{x})^2 + (y_i - \bar{y})^2] \qquad \text{Eq. 3.73}$$

Differentiating Eq. 3.71 with respect to $\bar{\mathrm{x}}$ yelds

$$\frac{\partial J}{\partial \bar{x}} = 2 \sum_{i=1}^{N} \{R^2 - [(x_i - \bar{x})^2 + (y_i - \bar{y})^2]\} \cdot 2(x_i - \bar{x})(-1) = 0 \qquad \text{Eq. 3.74}$$

or

$$\sum_{i=1}^{N}\{R^2-[(x_i-\bar{x})^2+(y_i-\bar{y})^2]\}x_i=\sum_{i=1}^{N}\{R^2-[(x_i-\bar{x})^2+(y_i-\bar{y})^2]\}\bar{x}=0 \qquad \text{Eq. 3.75}$$

Therefore,

$$R^2\sum_{i=1}^{N}x_i=\sum_{i=1}^{N}[(x_i-\bar{x})^2+(y_i-\bar{y})^2]x_i \qquad \text{Eq. 3.76}$$

Similarly, differentiating Eq. 3.71 with respect to $\bar{y}$ yelds

$$R^2\sum_{i=1}^{N}y_i=\sum_{i=1}^{N}[(x_i-\bar{x})^2+(y_i-\bar{y})^2]y_i \qquad \text{Eq. 3.77}$$

Eq. 3.73, Eq. 3.76 and Eq. 3.77 cam ne solved, even though quadratic, using the following tricks. First, let us simplify the notations using the following conventions. Summation over N indices are now as written as p_i, so that p_1 stands for $\sum_i x_i = x_1+x_2+\cdots+x_i+\cdots+x_N$ and

$$p_2=\sum_i {x_i}^2 \qquad p_3=\sum_i x_iy_i \qquad p_4=\sum_i y_i \qquad p_5=\sum_i {y_i}^2$$

$$p_6=\sum_i {x_i}^3 \qquad p_7=\sum_i x_i{y_i}^2 \qquad p_8=\sum_i {y_i}^3 \qquad p_9=\sum_i {x_i}^2y_i$$

With these conventions, Eq. 3.73, Eq. 3.76 and Eq. 3.77 become respectively

$$NR^2=p_2-2p_1\bar{x}+N\bar{x}^2+p_5-2p_4\bar{y}+N\bar{y}^2 \qquad \text{Eq. 3.78}$$

$$R^2p_1=p_6-2p_2\bar{x}+p_1\bar{x}^2+p_7-2p_3\bar{y}+p_1\bar{y}^2 \qquad \text{Eq. 3.79}$$

$$R^2p_2=p_9-2p_3\bar{x}+p_4\bar{x}^2+p_8-2p_5\bar{y}+p_4\bar{y}^2 \qquad \text{Eq. 3.80}$$

Eq. 3.78, Eq. 3.79 and Eq. 3.80 can be solved.
Multiply Eq. 3.78 by p_1, and subtract N multiplied by Eq. 3.79 to give

$$p_2p_1-Np_6-2\bar{x}({p_1}^2-Np_2)+p_1p_5-Np_7-2\bar{y}(p_1p_4-Np_3)=0 \qquad \text{Eq. 3.81}$$

Multiply Eq. 3.78 by p_1, and subtract N multiplied by Eq. 3.80 to give

$$p_2p_1-Np_9-2\bar{x}(p_1p_4-Np_3)+p_1p_5-Np_8-2\bar{y}({p_4}^2-Np_5)=0 \qquad \text{Eq. 3.82}$$

It is now possible to solve Eq. 3.81 and Eq. 3.82 for $\bar{x}$ and $\bar{y}$; the result can be written in matrix form as

$$\begin{pmatrix}a_1 & b_1\\ a_2 & b_2\end{pmatrix}\begin{pmatrix}\bar{x}\\ \bar{y}\end{pmatrix}=\begin{pmatrix}c_1\\ c_2\end{pmatrix} \qquad \text{Eq. 3.83}$$

where

$$a_1 = 2({p_1}^2 - Np_2) \qquad b_1 = 2(p_1p_4 - Np_3) \qquad c_1 = (p_1p_2 - Np_6 + p_1p_5 - Np_7)$$

$$a_2 = b_1 \qquad b_2 = 2({p_4}^2 - Np_5) \qquad c_2 = (p_2p_4 - Np_8 + p_4p_5 - Np_9)$$

from which

$$\bar{x} = \frac{c_1b_2 - c_2b_1}{a_1b_2 - a_2b_1} \qquad \bar{y} = \frac{a_1c_2 - a_2c_1}{a_1b_2 - a_2b_1} \qquad \text{Eq. 3.84}$$

Having solved $\bar{x}$ and $\bar{y}$ it's possible to substitute into Eq. 3.78 to get R

$$R = \sqrt{\frac{1}{N}(p_2 - 2p_1\bar{x} + N\bar{x}^2 + p_5 - 2p_4 + N\bar{y}^2)} \qquad \text{Eq. 3.85}$$

Eq. 3.84 and Eq. 3.85 can be used to calculate the coordinates of the center and hence the radius of the best fitting circle.

This algorithm has been implemented and tested in this current research work, thanks to the script already built-in in Matlab© and results of this fitting are shown in 6.2.

3.4 The Kalman filter

Kalman filter's basic intuition is the ability to update an estimate of least squares (e.g. due to the introduction of new observations) without recalculating the system as a whole.
Generally, when least squares are adopted, the starting point is the assumption that measures **y** are linearly dependent from the parameters **x**, so that the system is like the following:

$$Ax + a = y \qquad \text{Eq. 3.86}$$

and, in the case of real measurements y_0

$$Ax + a - y_0 = (y - y_0) = v \qquad \text{Eq. 3.87}$$

$$Ax - b = v \qquad \text{Eq. 3.88}$$

Assumed that

$$b = (y_0 - a) \qquad \text{Eq. 3.89}$$

This equation has to be solved by imposing a minimum condition:

$$v^T P v = min \qquad \text{Eq. 3.90}$$

as well as the condition that the estimates of y belong to the plane of eligible measures:

$$\hat{x} = (A^T PA)^{-1} A^T Pb = N^{-1} A^T Pb \qquad \text{Eq. 3.91}$$

The Ordinary Least Squares estimation (OLS) of the parameters x is a linear estimation of the observations y.
If new measures are added, (that we will call $y_{0(i+1)}$) it is expected that even for the new updated parameters estimate (called x_{i+1}) they still represent a linear mathematical application like the following

$$\hat{x}_{i+1} = L\hat{x}_i + Kb_{i+1} \qquad \text{Eq. 3.92}$$

It is assumed that the parameters x calculated at the i+1 updating step are linear functions of the parameters previously obtained and the new measures, added inside b_{i+1}.

The crucial problem is to compute the two matrices L and K.

By separating the parameters mentioned above in the previous step by their increase because of the new measures, another way to express this concept is:

$$\hat{x}_{i+1} = \hat{x}_i + K(b_{i+1} - A_{i+1}x_i) \qquad \text{Eq. 3.93}$$

Comparing Eq. 3.92 and Eq. 3.93, it can be derived that

$$L = (I - KA_{i+1}) \qquad \text{Eq. 3.94}$$

The only real unknown is then the K matrix, called Gain Matrix or Kalman Matrix.
R.E. Kalman solved this problem in 1960. Actually, Kalman obtained updating equations dealing with a more general problem and from a purely statistical point of view.

It is evincible that the solution can be updated, but not only that: it is also possible to update the parameters variance/covariance matrix and residuals one. These updates, with dynamic analogy, in which the system evolves over time, are called *epochs*.

The structure of the matrix design could be such as to only need updating measurements and parameters of the same period and/or only on the next one. In this case, the recursive solution needs to revers matrices of size equal to the non-null columns of the matrices A_{i+1}, with an evident computing time saving.

The recursive solution is convenient due to the bi-diagonal structure of the design matrix. At each epoch new measures and/or new parameters are added, which, however, involve only the parameters of the previous epoch. Due to this specific reason, the normal matrix will be a tri-diagonal matrix, such as the following; the terms in the matrix are written as scalar but can also represent matrices.

$$A = \begin{bmatrix} a_1^1 & & & & \\ a_2^1 & a_2^2 & & & \\ & a_3^1 & & & \\ & a_4^1 & a_4^2 & & \\ & & a_5^1 & & \\ & & a_6^1 & a_6^2 & \\ & & & a_7^1 & \end{bmatrix} \qquad N = \begin{bmatrix} n_1^1 & n_1^2 & \cdots & \cdots & \cdots & \cdots & 0 \\ & n_2^2 & n_2^3 & \cdots & \cdots & \cdots & 0 \\ & & n_3^3 & n_3^4 & 0 & \cdots & 0 \\ & & & \cdots & \cdots & \cdots & \cdots \\ & simm. & & & \cdots & \cdots & \cdots \\ & & & & & n_{N-1}^{N-1} & n_{N-1}^{N} \\ & & & & & & n_N^N \end{bmatrix} \qquad \text{Eq. 3.95}$$

In literature there are some other demonstrations based on different techniques (Casella, et al., 2001), (Farrell & Barth, 1998) and (Tiberius, 1998) but they are not autonomous: without the basic hypothesis of having a normal tri-diagonal matrix, (which means that parameters in epoch **k** only involves epochs **k-1**, **k** itself and/or **k+1)**, the recursive update would be impossible.
It is possible to solve the problem through the normal matrix factorization of two matrices:

$$Nx = n \qquad \text{Eq. 3.96}$$

$$LRx = n \qquad \text{Eq. 3.97}$$

where $L=R^T$. The shape of the two matrices is the triangular band like shown in Figure 3.5.

By inverting the matrix L we have:

$$Rx = L^{-1}n \qquad \text{Eq. 3.98}$$

From this expression it is possible to find the last value x_N, and then calculate the upgrade, with the following:

$$r_n x_n = -I_n^{-1} \times n \qquad \text{Eq. 3.99}$$

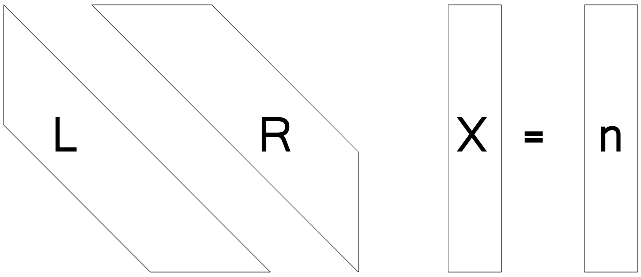


Figure 3.5 - Graphical Scheme of matrices factorization

Some authors demonstrate the Kalman filter as a particular case of sequential least squares; the demonstration is relatively simple, but unfortunately not rigorous. To understand better the diversity between the two methods should be noted that:

- sequential least squares were born as a "static" approach to the updating problem: each time new observations are added (or pseudo-observations), increasing the design matrix rows number, but usually not the number of columns;
- The Kalman filter, instead, is able to face with the opportunity that every time there are new unknown parameters, with a certain law dependent by the previous epoch parameters. The update problem in this case is to follow the time evolution of these parameters.

As already mentioned, the estimate sequential last epoch is the best one, (with reference to the L2 norm of all parameters involved), while the latest estimate of the filter represents the best one for the parameters related to L2 last epoch. To find the optimal values of the parameters on all epochs (even before epoch **i**), it's necessary to take into account also the new measures (after **i**) with a so-called *smoothing* operation, which is basically a backward replacement of the parameters.

Now the question could be: is the filter useful or essential?
As already said, Kalman filter is composed of two steps: filtering and smoothing. The first one is used to determine the best parameters estimate for the current epoch, the second one determines the best parameters estimate about all epochs; for this second operation the problem is the need of storing, all along the filtering, all parameters values and their dispersion matrix. Actually, if the observations in post-processing could be treated, the filter may not be necessary. In this case, it should be only necessary to store all the observations and, if possible, to treat them with least squares: the only remaining problem could be the storage size & memory needed to storage and solve a big equations system. However, the filter still remains essential when real time needings occur, like knowing and managing the position/speed of an object, to change the route or to correct also an instrumental drift.

Some instruments may provide data only within a certain measurement range, so if the flow conditions do not include drift corrections, this limit may be exceeded briefly; in this situation, only filtering could be enough. The deformation of a real time GNSS stations network deformation can be performed with the aid of the filtering only, while the position of an aircraft for a GPS/INS flight requires the best precision available and can be performed in real time, but necessarily also in post processing: by means of traditional least squares and smoothing and filtering operation.

It is important to emphasize the importance of filtering for the real time control. The ability to check at any time the consistency of the new measures with the data analyzed it is necessary condition for the removal of erroneous measurements or, in other cases, to highlight important displacements and deformations to be monitored.

Figure 3.6 - Barack Obama gives a 2008 National Medal of Science to Rudolf Kalman