



**HAL**  
open science

## Cartography For Cooperative Manoeuvres With Autonomous Land Vehicles

Javier Alonso, Vicente Milanés, Enrique Onieva, Joshué Pérez Rastelli, Carlos  
González, Teresa de Pedro

► **To cite this version:**

Javier Alonso, Vicente Milanés, Enrique Onieva, Joshué Pérez Rastelli, Carlos González, et al.. Cartography For Cooperative Manoeuvres With Autonomous Land Vehicles. the journal of navigation, 2011. hal-00737644

**HAL Id: hal-00737644**

**<https://inria.hal.science/hal-00737644>**

Submitted on 2 Oct 2012

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

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

# Cartography For Cooperative Manoeuvres With Autonomous Land Vehicles

Javier Alonso, Vicente Milanés, Enrique Onieva, Joshué Pérez,  
Carlos González, Teresa de Pedro

(*Centro de Automática y Robótica (CAR), UPM-CSIC, Madrid*)  
(Email: javier.alonso@car.upm-csic.es)

This article presents a cartographic system to facilitate cooperative manoeuvres among autonomous vehicles in a well-known environment. The main objective is to design an extended cartographic system to help in the navigation of autonomous vehicles. This system has to allow the vehicles not only to access the reference points needed for navigation, but also noticeable information such as the location and type of traffic signals, the proximity to a crossing, the streets en route, etc. To do this, a hierarchical representation of the information has been chosen, where the information has been stored in two levels. The lower level contains the archives with the Universal Traverse Mercator (UTM) coordinates of the points that define the reference segments to follow. The upper level contains a directed graph with the relational database in which streets, crossings, roundabouts and other points of interest are represented. Using this new system it is possible to know when the vehicle approaches a crossing, what other paths arrive at that crossing, and, should there be other vehicles circulating on those paths and arriving at the crossing, which one has the highest priority. The data obtained from the cartographic system is used by the autonomous vehicles for cooperative manoeuvres.

## KEY WORDS

1. Cartographic System,
2. Autonomous Vehicles,
3. Intelligent Transportation Systems (ITS).

1. INTRODUCTION. Cartography is one of the most important tools for a driving system. Not only does it contain the reference points that vehicles can use to remain in their lane but it also allows drivers to know in advance the junctions, roundabouts or even sharp bends and, with the help of a communications system, other vehicles in the area which might be relevant to planning manoeuvres. In addition, by adding information to the different zones, one can know the road signs at a junction, a roundabout or part of a driving area.

This work has been done for the AUTOPIA project. The Robotics and Automation Centre (CAR), formerly known as the Industrial Automation Institute (IAI), part of the Spanish Superior Scientific Research Council (CSIC), started the AUTOPIA project in 1996. This research group has been working toward the full



Figure 1. CAR Autonomous Vehicles.

automation of autonomous vehicles in urban areas and is focusing now on the cooperative manoeuvres between several cars. The research centre has developed three dual mode cars (Figure 1) and has an experimentation track that looks like an urban area. The cars are fully automated but can be also operated manually. The accelerator, the brake, and the steering wheel of the vehicles have been automated and are controlled by a computer. The vehicles are also equipped with Differential GPS, Inertial Measurement Units (IMU), Magnetic Compass, and Wheel Encoders to provide a precise positioning (Milanés et al., 2008). The car position is compared with the map of the desired route and a fuzzy controller sends the required orders to the accelerator, brake, and steering wheel to keep the car on its route (Onieva et al., 2009).

From 2005 to 2008 CAR group led the Control Algorithms package of the European project “*EU STREP FP6-028062 CyberCars2: Close communications for Cooperation between CyberCars*” (CC2). A common architecture and a communications standard were developed to allow cooperation between the consortium partners: TNO (Holland), INRIA (France), CRF (Italy) and CAR (Spain) (Naranjo et al., 2009). But, to perform cooperative manoeuvres among several autonomous vehicles, a new cartographic system was also needed (Alonso, 2009). The main reason for redesigning the cartographic system was to facilitate the study of road junction priority signs. To establish road junction priorities, an extended cartographic system was necessary to provide additional information, such as the location and type of road signs, proximity of a junction and the actual street junctions.

For standalone navigation, just the coordinates of the way-points en-route are needed. But when several cars are required to cooperate with each other more information is needed. In order to get this, the maps were modified to be able to determine, for instance, in which streets the other cars were, or which streets lead toward an intersection or to a roundabout. The cartographic system was changed into a database containing the topological information (connections among the features of the map), the metrical information (coordinates) and other information such as the names of the streets, the maximum driving speed, etc. (Zhao, 2001) (Drane and Rizos, 1997) (White, 1993).

2. RELATED WORK. Nowadays, navigation map databases contain much more information than is needed for navigation purposes. This additional information can be used by Advanced Driving Assistance Systems (ADAS) (Rowell, 2001), but it can also be used by autonomous cars to perform complex manoeuvres. Several firms provide cartographic systems for commercial navigators, but these maps provide data only for use by a human driver. We now review some cartographic systems used for autonomous navigation by some well-known groups.

The DARPA Grand Challenge was one of the most relevant events for autonomous cars and it marked an inflexion point in these subjects. For instance EUREKA Prometheus Project (1987–1995) gave the starting shot for autonomous driving in Europe but it was the DARPA Grand Challenge that gave the urge to the topic. Autonomous navigation was not considered a noteworthy topic until this event made people realize that automatic navigation was not just science fiction and could be achieved. In the 2005 edition, the goal to complete a substantial off-road course within a limited time was achieved. A pre-defined route of way-points was travelled through by the cars using GPS and other self-perception sensors to get their approximate positioning, and by use of Artificial Vision, Lidar, etc., to keep the car on the road and to avoid obstacles (Thrun, 2006) (Thrun et al., 2006).

The DARPA 2007, “Urban Challenge” has been the latest competition. This competition required the autonomous vehicles not only to be able to travel along a pre-established route, they had to be able to drive through an urban environment with traffic, and perform such complex manoeuvres as overtaking and crossing intersections. Several autonomous vehicles and many research groups were involved in the DARPA “Urban Challenge”. To explain the navigation techniques of all the autonomous vehicles involved is outside the scope of this article, but to explain the most successful ones can be a great example of the navigation techniques used by autonomous vehicles nowadays.

DARPA provided all teams with a route network definition file (RNDF), which defined the road positions, number of lanes and intersections in GPS coordinates. Several lists of checkpoints that the vehicle was required to go through were also provided to perform the evaluation of the cars. Based on the minimal reliance on GPS as observed in previous editions, the MIT focused their efforts on perception-based autonomous driving. So, the information of the digital map was used to aid the drivability map used to navigate. This map is built by fusing the sensor information. The information provided by the RNDF was mainly used by the motion planner to set a goal for the navigator (Leonard et al., 2009).

Contrary to the MIT opinion, the Carnegie Mellon University research group trusted in GPS. They managed to get 0.3 metres error in position estimation with GPS and differential corrections, and even after driving 1 km without the GPS, the Inertial Measurement Unit (IMU) and the wheel encoders managed to keep the error below 0.9 metres. As the lane geometry was not *a priori* known to metre accuracy, they had to combine the position estimation system and the measurements of road lane markers (Urmson et al., 2008a) (Urmson et al., 2008b). To generate the mission plan needed for navigation, the data provided in the RNDF was used to create a graph that encoded the connectivity of the environment. Each waypoint in the RNDF became a node in this graph, and directional edges (representing lanes) were inserted between waypoints and all other waypoints that they could reach. For instance, a waypoint defining a stop line at an intersection would have edges connecting it to

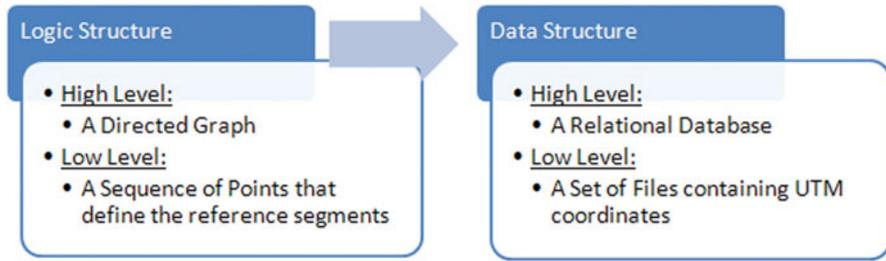


Figure 2. Logic and Data structure.

all waypoints leaving the intersection that could legally be driven to. But the cartographic system was not only used for navigation purposes, it was also used to provide additional information for the intersections, roundabouts, etc. The decision-making system that controls the crossing of intersections also got support from the cartographic system (Seo and Urmsom, 2008). The automated car took information about the regions of interest from the RNDF and, according to it, set the observation areas to where the car sensors were to be focused.

In Europe there are also autonomous vehicles research groups that are working towards urban navigation. The Institut National de Recherche en Informatique et Automatique (INRIA) has been working in autonomous cars since 1994 (Bouraoui et al., 2006). In a joint effort with the Swiss Federal Institute of Technology in Zürich (ETH Zürich), they presented a new cartographic and navigation system for INRIA CyberCars in (Macek et al., 2008). They use the *a priori* knowledge of the environment to select some reference points for the autonomous car to follow. A graph is used for the cartographic system, where the nodes are the reference points to be followed and the arcs that join them contains the distance between the nodes. The car can quickly select the shortest route by using the information from the arcs.

**3. THE CARTOGRAPHIC SYSTEM.** The philosophy of the GPS Navigators has been followed when designing the new upgraded cartography. The decision taken was to create a cartographic structure in two levels (see Figure 2). The high-level structure is a directed graph that relates arcs and nodes. The arcs represent the reference lines that the vehicle uses to navigate, and the nodes are points where at least an arc of entry and an arc of exit meet. A node may have several points of entry and exit and it contains the internal paths that connect the points of entry and exit within the node. These will be named ‘inner paths’. These paths provide a comprehensive description of all the possible paths that a vehicle can take from any point of entry to the node to any possible exit (Figure 3). The low level structure contains the sequence of points that define the reference segments that the vehicle will go over. These points are stored in a series of files indexed by the high-level structure.

The data structure also has two levels. The low level is pure cartography, that is, a set of files containing UTM (Universal Transverse Mercator) coordinates of the significant points for the paths of the arcs and also the internal paths in the nodes. The number of points stored must be enough to describe any route as a series of straight stretches without significant error. When a straight segment is defined it will

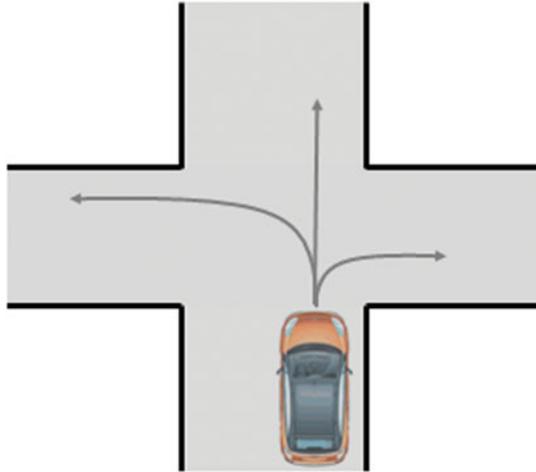


Figure 3. Representation of all the possible routes of a vehicle at a junction (called “Paths” in the database).

be sufficient to store the points of the ends, whereas for curved segments a point will be necessary at certain intervals. At the high level the data structure is a relational database. The nodes of the graph (i.e. junctions, roundabouts and other specially significant points of the driving zone) are a table in the database, the arcs (i.e. streets or ways) are another of the tables and finally, the last table contains inner paths. Each inner path is linked to the node to which it belongs, and each node is also linked to its own inner paths. In this way given an inner path we can find the node to which it belongs, and given a node we can access all its inner paths. Each inner path, as well as each arc is linked to the low level file containing the coordinates of the points defining it.

Figure 3 shows the possible paths inside a crossing represented as red lines. Consequently, the node that would represent this crossing will contain the necessary information so the vehicle can reach any point of exit from any point of entry of the node. This data structure permits a vehicle to select one of the internal paths to the node upon reaching a junction. The name of the file that contains the right path will be obtained from the database, thanks to a list of nodes by which we want the vehicle to keep moving.

3.1. *Arcs and Nodes.* The arcs represent street lanes. In other words, in a two-way street there will be two arcs, one in each direction. The nodes represent those places to be distinguished for any reason, perhaps because of the presence of a road sign, narrowing of the street, pedestrian crossing or street junction. A necessary condition for nodes is that there is at least one arc of entry and one arc of exit. Junctions and roundabouts are represented as nodes. This representation facilitates the implementation of algorithms of coordination of movements at these points. We also chose to represent curves as nodes for practical purposes. If vehicles must turn 90 degrees within a small space, we have to pay attention to these points, reducing speed and assuring the coordination of vehicles. In addition, by considering the curves as nodes, we can use the automatic point generation program which will be discussed in the next section.



Figure 4. Aerial photograph of the CAR driving area.

3.2. *Internal paths within nodes.* The internal paths to the node represent the possible routes that can be taken at a junction, roundabout, bend or any specific point within a driving area. Paths represents all the routes that connect entry points to exit points within a node.

3.2.1. *Design of the internal paths of nodes.* The tracking control system works essentially on straight segments (Garcia et al., 2002). The idea is to turn curved sections into a series of straight segments. The design of these curved sections is carried out by using clothoids (Knowles, 2006) and circular curves. Thus the system designs a route as a series of straight and curve segments, each curved section planned as a gentle curve with a progression of constant curvature. Later this gentle curve is transformed into small straight sections. To turn a curved section into a series of small straight segments a sampling is performed at 0.5 metres intervals. These values are stored in a vector of variable length and transferred to a file to be accessed from a database.

3.3. *Representation of the CAR Driving Area.* The driving area (ZOCO) is the test track for stand-alone vehicles of the CAR (See Figure 4). It is used exclusively for research tasks, therefore, there is no traffic of people or vehicles. The street distribution simulates blocks in a city and six-metre-wide, two lane streets. The graph that represents the driving area is shown in Figure 5. The junctions are represented as nodes ( $N_i$ ), the streets as arcs ( $A_i$ ) and the paths within the junction as ( $C_i$ ). As can be seen in Figure 5, each node contains one roundabout ( $C_1$  in  $N_1$ ) and two or more internal paths (curves in  $N_4$ ,  $N_5$ ,  $N_8$  and  $N_9$ ) depending on the type of node. These paths join the entry points to the exit points of the node. So, a node that represents a sharp bend will have only two possible paths, one in each direction. However a crossing with two streets can have 12 ways of going through. All the paths that a vehicle may take within the node (12 in the case of a crossing) are stored (See Figure 6).

3.4. *Signposting in the Driving Area.* If the right-of-way of each vehicle at an intersection is determined by road signs, it will be necessary to know those signs existing. There are different ways to identify such signs. One can detect the present

Legend:

Black: Nodes (N1..N9)

Blue: Arcs (A1..A22)

Red: Path within the junctions (C1..C39)

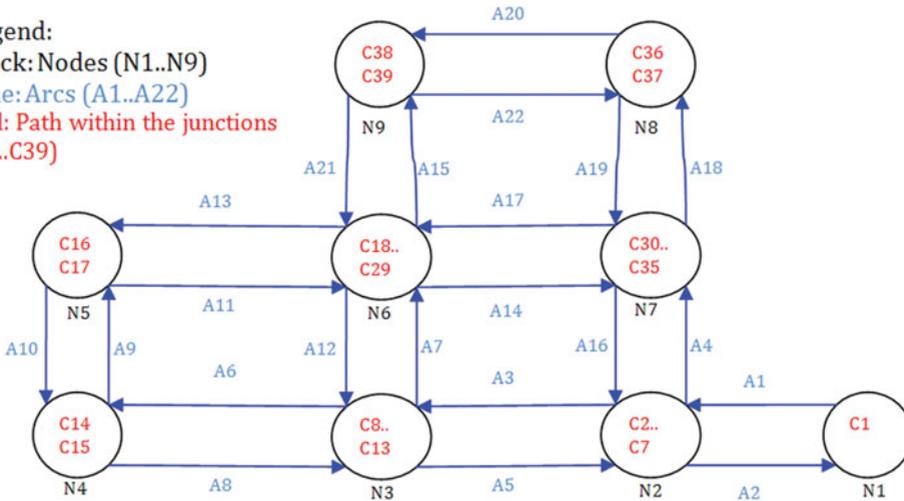


Figure 5. The new cartographic system graph.

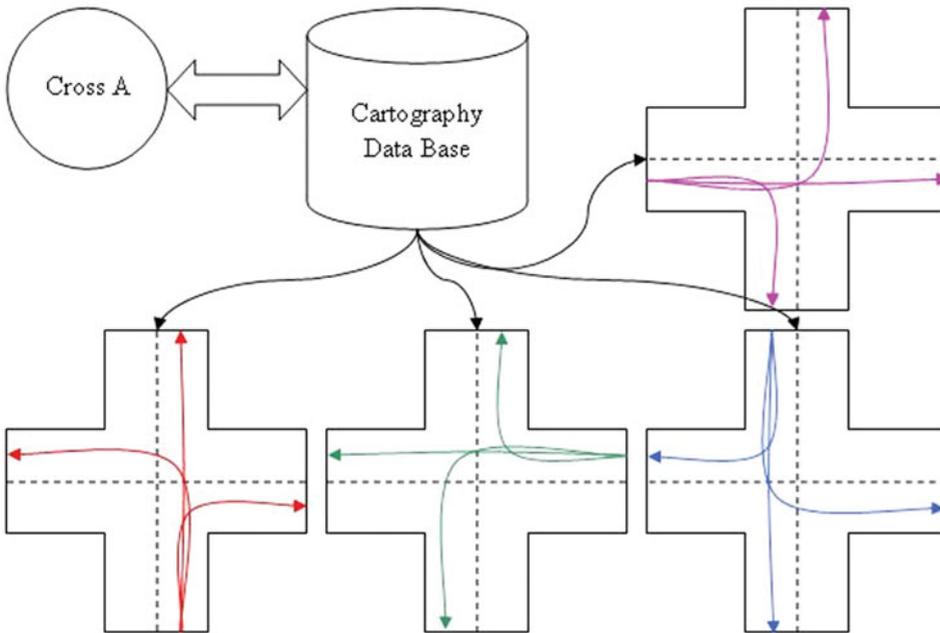
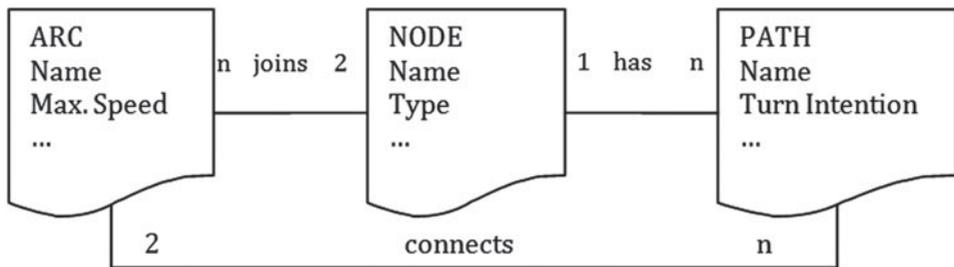


Figure 6. The 12 possible paths that may be taken at a typical junction in x.

signs by using wireless sensor networks (Milanés et al., 2010) or computer vision cameras (Garcia-Garrido et al., 2006), or by simply noting down the presence of such signs as additional data on the maps of vehicles. The signs that change the right of way at an intersection are: Priority Road, Give Way and Stop. Since it may be necessary to include additional information in the map, one can leave the door open for adding new signs on to each stretch, such as speed limits, no overtaking, etc.

Table 1. Information contained in the database.

Node	Arc	Path
Node identifier	Arc identifier	Path identifier
Type of node	Maximum speed of the arc	Node to which it belongs
List of all related paths	Starting node	Arc from which it starts
List of outgoing arcs	Target node	Arc which it reaches
List of incoming arcs	List of UTM coordinates of the street	Turn Direction: Turn Right, turn Left or keep Straight



ARC	NODE	PATH
Name (PK)	Name (PK)	Name (PK)
Max. Speed Limit	Type	Node_Name (FK)
Initial_Node (FK)		Initial_Arc (FK)
Final_Node (FK)		Final_Arc (FK)
UTM_Coordinates		Turn Direction
		UTM_Coordinates

Figure 7. Relational diagram of the database (top) with the breakdown of associated information (bottom).

3.5. *Cartographic System Database.* The method used for storing information is a relational database (See Figure 7). This structure allows one to save simultaneously the topology of the zone of the driving area and the names of the files that contain the UTM coordinates of the segments that will make up the necessary maps for the driving. The information has to be distributed among the different entities of the cartographic system. The database guarantees access to any data item from another. For instance given an arc one can access the node of destination or the node of origin (see Table 1). The use of a relational database avoids storing duplicated information, and allows the formation of the strong relationship among nodes, arcs and paths. The relationships that have been included in the database are:

- A directed arc “joins” two nodes, in other words, there is an initial node and a final one.
- A directed inner path “connects” two arcs, initial and final.
- And each node “has” a series of associated paths.

In addition to the files that contain the UTM coordinates of each arc and path, some additional information has been included in the nodes, arcs and paths for better identification. The maximum speed limit has been added to each straight section of the circuit, in other words, for each lane (arc). To facilitate manoeuvres at the junctions (nodes), the type of node is also added (curve, roundabout, T-junction or X-junction). With respect to each possible path within a junction (internal path to the node), the direction in which the path turns; right, left or straight is also added.

4. THE NAVIGATION SYSTEM. This system is used to define the course of the autonomous cars. A course is defined as a series of nodes through which one wants the vehicle to pass, and whose coordinates will be searched in the cartography of the zone at run-time. The initial position of the vehicle and the next target node determine the reference path unambiguously. The execution may be defined as follows: the car starts by looking for the nearest arc and goes through it until it approaches the node where the arc finishes. The starting point of the course might be at any point of the defined route. The navigation system searches for the arc that contains the starting point by going through the arcs and examining its trajectories. Then the reference points of the arc are downloaded and used by the car controller to perform the longitudinal and lateral control until the next node is reached. By knowing the node it is approaching, the arc that it reaches and the next node, it selects the right inner path, downloads its reference points and makes the car track them. Once the last point of the inner path is reached, it downloads the arc that leads to the following node and the procedure is repeated until the last node is reached. The algorithm, according to the actions that have to be done at different moments is:

- *Initially:*
  - At the beginning the track is defined as a set of consecutive nodes. The vehicle may start at any point.
  - Query the database which arc joins the first and second nodes, and to compare it with the arc where the vehicle is located exactly. If it does not coincide, repeat the process until the arc where the vehicle is located is found.
  - Open the file of the arc.
  - Download the coordinates of the arc.
  - Make the car go through the reference points until the next node is reached.
- *Upon reaching a node:*
  - Query the database which arc joins the current and the next nodes.
  - With the next arc and the one we have just finished, ask which path joins them within the node.
  - Download the coordinates of the inner path.
  - Make the car go through the reference points until the next arc is reached.
- *When an inner path ends:*
  - Open the file of the next arc.
  - Download the coordinates of the arc.
  - Make the car go through the reference points until the next node is reached.

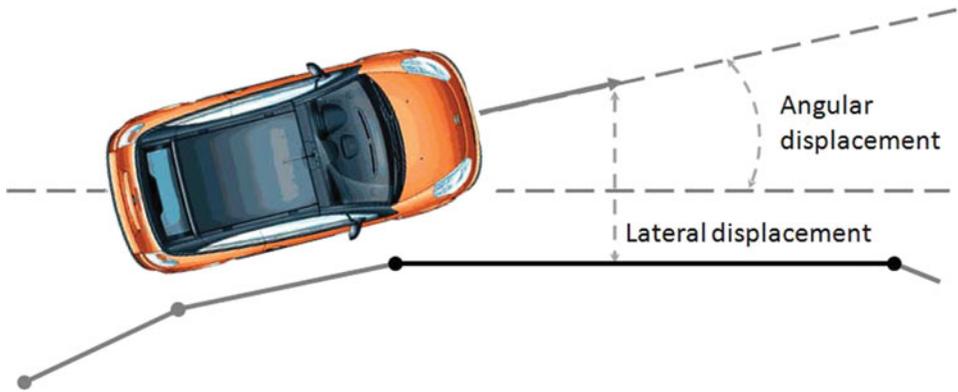


Figure 8. Lateral and angular deviation over the reference segment.

5. **AUTONOMOUS VEHICLES.** The autonomous vehicle used in the experiments is presented in this section. AUTOPIA's new cartographic system is placed in a Citroen C3 Pluriel that is able to perform autonomous driving. The vehicle is equipped with a double-frequency GPS receiver running in RTK carrier phase differential mode that supplies 2 cm resolution positioning at a refresh rate of 5 Hz. To prevent the degradation of positioning when the correction signal is lost, the vehicle uses its own odometry, supplied by a set of built-in sensors in the wheels whose measurements can be read by accessing the Controller Area Network (CAN) bus of the vehicle; and an inertial measurement unit (IMU) Crossbow IMU300CC placed close to the centre of the vehicle. Using this system, automated vehicle guidance is possible in city areas. We tested our system for up to 5 minutes before the drift became too high. An onboard computer uses the accurate position and the speed provided by the wheel encoders to send control signals to the steering wheel, accelerator and brake. The computer uses a fuzzy logic controller to calculate the correct goal positions of the car actuators (Naranjo et al., 2007) (García and de Pedro, 1998).

The devices that make it possible to act on the throttle and brake of the car are an electro-hydraulic system capable of injecting pressure into the car's braking system, and an analogic output card that is used to send signals to the car's internal engine computer to demand acceleration or deceleration. For the steering wheel, a power signal is sent to the Electric Power Steering (EPS) motor bypassing the EPS sensor and its control card (Perez et al., 2009).

Basically, there are two fuzzy controllers that manage both vehicle control signals: speed and steering, also known as longitudinal and lateral control. The longitudinal control manages the throttle and brake, whilst the lateral control manages the steering wheel. Steering fuzzy control is in charge of minimizing the trajectory deviations. The inputs of this controller are the lateral and angular deviation over the reference segment (see Figure 8). The output is the angle that the steering wheel must be turned to correct the trajectory deviation. Throttle and brake fuzzy control permit the vehicle to adapt its speed to a reference, reducing or increasing this speed when necessary in order to maintain its route or to maintain a safe distance to the preceding

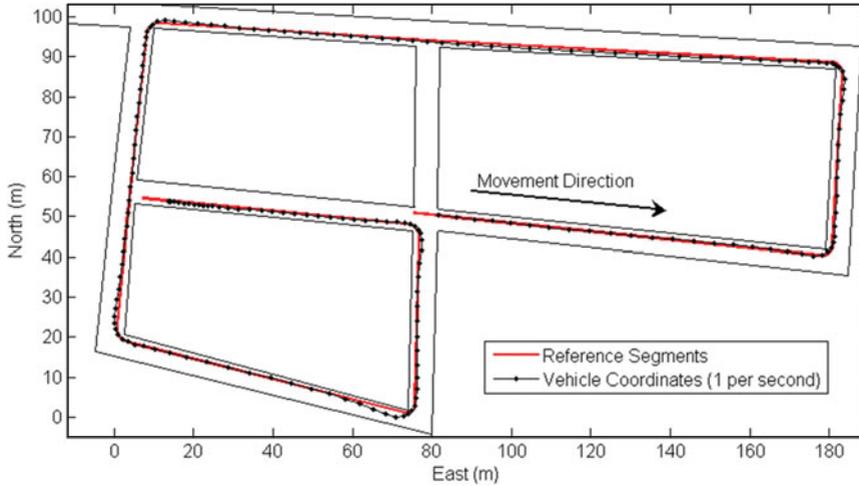


Figure 9. Vehicle trajectory and the reference lines followed.

vehicle. The inputs of these controllers are the difference between the desired and the present speed, the acceleration, and, if the vehicle is performing an Adaptive Cruise Control (ACC) manoeuvre, the distance to the preceding vehicle and the variation of this distance. The outputs of these controllers are the pressure that must be applied over each pedal in order to minimize the speed errors.

The fuzzy controllers are based on a fuzzy inference machine named ORBEX (Spanish acronym for Fuzzy Experimental Computer), which was developed at the CAR (García and de Pedro, 2000). ORBEX allows the definition of the autonomous vehicle behaviour by simple IF-THEN rules that are easily understandable. The details of the fuzzy inference machine are: the t-norm minimum and the t-conorm maximum are used to implement *and* and *or* operators, respectively; the Mamdani-type inference (Mamdani, 1976) is used, and the defuzzification operator is the centre of mass.

**6. EXPERIMENTS.** The objective of this experiment was to show the operation of the new cartographic system. A route was established by choosing a list of nodes to go through. The archives containing the reference point coordinates of each new arc/segment were loaded when the vehicle was near the end of the current arc/segment. When the last reference point of a given segment was 6 metres ahead (the distance from the GPS receiver to the front of the car is almost 4 metres and the look-ahead distance was set to 2 metres) the system loaded a new archive and took the first reference point of the new archive. Figure 9 shows the Northing and Easting UTM coordinates as black dots with respect to the origin (at 458874.86E, 4462461.00N) and the reference segments as pale gray lines.

Figure 9 shows the behaviour of the new cartographic system, but the scale of the figure does not allow appreciation of when the reference points are loaded. To be able to follow the operation of the new cartographic system, Figure 9 has been expanded

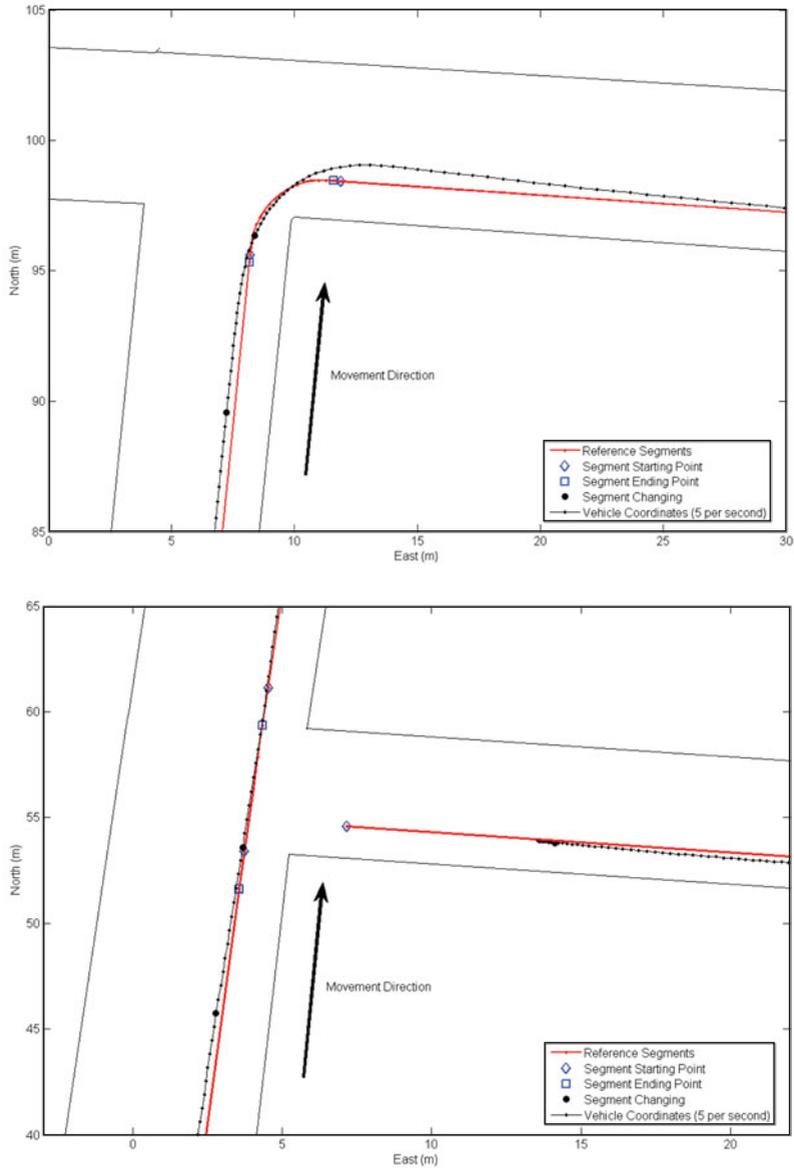


Figure 10. (Top) Detailed vehicle trajectory and the reference lines followed in a right turn. (Bottom) Detailed vehicle trajectory and the reference lines followed in crossing where the vehicle continues ahead.

at a right turn in an intersection (Figure 10 (top)), and at an intersection where the vehicle follows straight line segment (Figure 10 (bottom)).

The reference points (red) marked with a rhombus represent the first point of a new reference segment, loaded from a new file. It is the first point of a new arc or path. Whereas the reference points marked with a square represent the last point of a section, be it a arc or an inner path, and also the last point of the file. The black points

shown in the figure are UTM coordinates provided by the GPS respect to the above mentioned position when the new file is downloaded. Figure 10 (bottom) shows the reference point change in a crossing where the vehicle has to go straight ahead. Five reference points, separated by 1.5 metres, have been downloaded representing the selected inner path in the node.

The experiment shows that the cartographic system allows the vehicle to move around the testing area and also shows how the different segments of the map are loaded. Consequently, the validity of the proposed system is demonstrated, allowing its use for the implementation of the cooperation algorithms at junctions.

**7. CONCLUSION.** A graph based cartographic system for autonomous car navigation has been presented. It has the capability to store a great amount of information including speed limits and traffic signals. The system structure would permit real time updating. The main reason to design this system was to allow the definition of junctions in order to facilitate the solution to priority conflicts as well as obtaining the trajectories in order to traverse them. The communications system will permit the car to calculate its priority level in order to know whether it has to give way to the other vehicles in, or approaching a node.

The proposed solution is based on the same ideas as the Carnegie Mellon University, INRIA or ETH Zürich research groups solutions. The main difference between them is what they are wanted for. The ETH Zürich/INRIA goal is to find the minimum distance route from the current position to the final destination. So, the emphasis of the graph is to find out the distance between the different reference points. The Carnegie Mellon University interest relies on the lanes and the waypoints that can be reached from other waypoints in order to find alternative routes for blockades and other obstacles. Our goals are to be able to determine the trajectories for autonomous vehicles and to define the special places such as junctions and sharp bends where automatic control could solve the traffic problems that might arise. The case we have considered specifically is how to determine the car behaviour at intersections depending on the existing traffic.

#### ACKNOWLEDGMENTS

The authors express their gratitude to TRANSITO (TRA2008-06602-C03-01) and City-Elec (PS-370000-2009-4) projects from Plan Nacional, GUIADE (P 9/08) project from Ministerio de Fomento and MARTA project (CENIT-20072006).

#### REFERENCES

- Alonso, J. (2009). Cooperative Driving Manouevres among Autonomous Vehicles. PhD thesis. Facultad de Informática, Universidad Politécnica de Madrid.
- Bouraoui, L., S. Petti, A. Laouiti, T. Fraichard and M. Parent (2006). Cybercar cooperation for safe intersections. In: Proc. IEEE Intelligent Transportation Systems Conference ITSC '06. 456–461.
- Drane, C. and C. Rizos (1997). Positioning Systems in Intelligent Transportation. Artech House ITS Series. Artech House Publishers.
- García, R. and T. de Pedro (1998). Modeling a fuzzy coprocessor and its programming language. *Mathware and Soft Computing* 5(2–3), 167–174.

- García, R. and T. de Pedro (2000). First application of the orbex coprocessor: Control of unmanned vehicles. EUSFLAT-ESTYLF Joint Conference. *Mathware and Soft Computing* 7(2–3), 265–273.
- García-Garrido, M. A., M. A. Sotelo and E. Martm-Gorostiza (2006). Fast traffic sign detection and recognition under changing lighting conditions. In: Proc. IEEE Intelligent Transportation Systems Conference ITSC '06. 811–816.
- García, R., T. de Pedro, J. E. Naranjo, J. Reviejo and C. Gonzalez (2002). Frontal and lateral control for unmanned vehicles in urban tracks. In: Proc. IEEE Intelligent Vehicle Symposium. Vol. 2. 583–588.
- Knowles, D. (2006). Real time continuous curvature path planner for an autonomous vehicle in an urban environment. Technical report. California Institute of Technology.
- Leonard, J., J. How, S. Teller, M. Berger, S. Campbell, G. Fiore, L. Fletcher, E. Frazzoli, A. Huang, S. Karaman, O. Koch, Y. Kuwata, D. Moore, E. Olson, S. Peters, J. Teo, R. Truax, M. Walter, D. Barrett, A. Epstein, K. Maheloni, K. Moyer, T. Jones, R. Buckley, M. Antone, R. Galejs, S. Krishnamurthy and J. Williams (2009). A perception-driven autonomous urban vehicle. In: The DARPA Urban Challenge (Springer Berlin/Heidelberg, Ed.). Vol. 56 of Springer Tracts in Advanced Robotics. 163–230. Springer-Verlag.
- Macek, K., D. Vasquez, T. Fraichard and R. Siegwart (2008). Safe vehicle navigation in dynamic urban scenarios. In: Proc. 11th International IEEE Conference on Intelligent Transportation Systems ITSC 2008. 482–489.
- Mamdani, E. H. (1976). Application of fuzzy logic to approximate reasoning using linguistic synthesis. In: Proceedings of the sixth international symposium on Multiple-valued logic. IEEE Computer Society Press. 196–202.
- Milanés, V., E. Onieva, B. Vinagre, C. González, J. Pérez and Alonso J. (2010). Sistema de asistencia a la conducción basado en una red de comunicaciones de bajo coste. DYNA.
- Milanés, V., J. E. Naranjo, C. Gonzalez, J. Alonso and T. de Pedro (2008). Autonomous vehicle based in cooperative GPS and inertial systems. *Robotica* 26(5), 627–633.
- Naranjo, J. E., L. Bouraoui, R. Garcia, M. Parent and M. A. Sotelo (2009). Interoperable control architecture for cybercars and dual-mode cars. *IEEE Transactions on Intelligent Transportation Systems* 10(1), 146–154.
- Naranjo, J. E., M. A. Sotelo, C. Gonzalez, R. Garcia and T. de Pedro (2007). Using fuzzy logic in automated vehicle control. *IEEE Intelligent Systems* 22(1), 36–45.
- Onieva, E., J. Alonso, J. Pérez, V. Milanés and T. de Pedro (2009). Autonomous car fuzzy control modeled by iterative genetic algorithms. In: Proc. 2009 IEEE International Conference on Fuzzy Systems. Korea. 1615–1620.
- Perez, J., C. Gonzalez, V. Milanés, E. Onieva, J. Godoy and T. de Pedro (2009). Modularity, adaptability and evolution in the autopilot architecture for control of autonomous vehicles. In: Proc. IEEE International Conference on Mechatronics ICM 2009. 1–5.
- Rowell, J. M. (2001). Applying map databases to advanced navigation and driver assistance systems. *The Journal of Navigation* 54, 355–363.
- Seo, Y. W. and C. Urmson (2008). A perception mechanism for supporting autonomous intersection handling in urban driving. In: Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS 2008. 1830–1835.
- Thrun, S. (2006). Winning the darpa grand challenge: A robot race through the mojave desert. In: Proc. 21st IEEE/ACM International Conference on Automated Software Engineering ASE '06. 11–11.
- Thrun, S., M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Stroh, C. Dupont and etc. (2006). The robot that won the darpa grand challenge. *Journal of Field Robotics* 23, 661–692.
- Urmson, C., D. Duggins, T. Jochem, D. Pomerleau and C. Thorpe (2008a). From automated highways to urban challenges. In: Proc. IEEE International Conference on Vehicular Electronics and Safety ICVES 2008 6–10.
- Urmson, C., J. Anhalt, H. Bae, J. Bagnell, C. Baker, R. E. Bittner, T. Brown, M. N. Clark, M. Darms, D. Demitrish, J. Dolan, D. Duggins, D. Ferguson, T. Galatali, C. M. Geyer, M. Gittleman, S. Harbaugh, M. Hebert, T. Howard, S. Kolski, M. Likhachev, B. Litkouhi, A. Kelly, M. McNaughton, N. Miller, J. Nickolaou, K. Peterson, B. Pilnick, R. Rajkumar, P. Rybski, V. Sadekar, B. Salesky, Y. Seo, S. Singh, J. M. Snider, J. C. Struble, A. Stentz, M. Taylor, W. L. Whittaker, Z. Wolkowicki, W. Zhang and

- J. Ziglar (2008b). Autonomous driving in urban environments: Boss and the urban challenge. *Journal of Field Robotics* **25**(8), 425–466.
- White, M. (1993). Digital maps – a fundamental element of ivhs. *Journal of Intelligent Transportation Systems* 1(2), 135–150.
- Zhao, Y. (2001). *Vehicle Location and Navigation Systems*. Artech House ITS Series. Artech House Publishers.