The NARVALO project: real time collision avoidance system in a GIS environment based on precise GNSS positioning

R. Marzocchi*
M. Leotta**
B. Federici*
G. Delzanno**

* Dip. di Ingegneria Civile, Chimica e Ambientale (DICCA), Università degli Studi di Genova, Italy
** Dip. di Informatica, Bioingegneria, Robotica e Ingegneria dei Sistemi (DIBRIS), Università degli Studi di Genova, Italy

NARVALO is a collision avoidance system for logistic platform areas that detects possible dangerous situations, which could lead to accidents, and consequently alerts vehicles and people involved. The system is based on the real time knowledge of precise position, speed and acceleration of the monitored vehicles and operators using ad-hoc GNSS (Global Navigation Satellite Systems) – IMU (Inertial Measurement Unit) receivers, core of the "NARVALO box". In this paper, we focus on the prototype of the control center developed for the real time evaluation of the collision risks. It consists of a Geographical Information System (GIS) collision avoidance algorithm, developed using geospatial open source C libraries (GDAL and OGR), and a server, developed using the C programming language, responsible for the scheduling of the GIS algorithm, the management of real time communications with the mobile devices and the storage of historical data in a remote geoDataBase (geoDB).
***Keywords**: collision avoidance system, precise GNSS positioning, real time, GIS, geodatabase.*

Il progetto NARVALO: sistema di allerta in tempo reale del rischio collisione in ambiente GIS basato sul posizionamento GNSS di precisione. *NARVALO è un sistema di allerta anti-collisione all'interno di aree interportuali, che individua situazioni potenzialmente pericolose che potrebbero originare incidenti, ed allerta mezzi e persone coinvolte. Il sistema si basa sulla conoscenza in tempo reale del posizionamento preciso, della velocità ed accelerazione di veicoli e persone dotate di appositi ricevitori GNSS (Global Navigation Satellite Systems) – IMU (Inertial Measurement Unit), cuore del "NARVALO box". In questo contributo si intende presentare principalmente il prototipo di centro di controllo in tempo reale sviluppato per la valutazione del rischio di collisione. Esso consiste in un algoritmo anti-collisione, sviluppato in ambiente GIS (Geographical Information System) mediante librerie C geospaziali open source (GDAL e OGR), e da un server, sviluppato in linguaggio C, responsabile della pianificazione dell'algoritmo GIS, della gestione della comunicazione in tempo reale con i dispositivi mobili e della memorizzazione di dati storici nel geoDataBase (geoDB).*
***Parole chiave**: sistema anti-collisione, posizionamento GNSS di precisione, tempo reale, GIS, geoDataBase.*

1. Introduction

NARVALO is a research project co-funded by the Italian Space Agency (ASI) involving two industrial partners, 'Optisoft srl' and 'Gter srl Innovazione in Geomatica, GNSS e GIS', and the University of Genova as academic partner. As the Italian acronym says (NAvigazione satellitare di precisione per la sicuRezza e la Viabilità in Aree LOgistiche), it consists in a collision avoidance system, capable of preventing collisions between vehicles and human operators (in the following called "humans") in a well delimited geographical area, i.e. logistic terminals for intermodal cargo containers.

In the automotive field, support systems based on emitters (lasers and cameras) and collision avoidance systems able to activate risk mitigation actions (e.g. braking the vehicle) are already marketed. Recently, Xueming *et al.* (2016) presented a novel brake behavior detection method relying on a color camera fixed on the windshield and able to capture the brake behavior of the front vehicle. Deng-Yuan *et al.* (2017) implemented a driver assistance system for vehicles detection and inter-vehicle distance estimation using a single-lens video camera on urban/suburb roads. Kolk *et al.* (2016) presented a collision mitigation system integrated in an advanced driver assistance system, aimed at reducing the impact speeds of vehicles by means of laser sensors (measuring range or opening angle) and various response strategies.

Real time collision detection systems are also deeply studied and implemented for highly dynamic applications such as 3D games, virtual reality, and physical simulators (Ericson, 2004).

In confined areas, such as airports, ports and logistic platform areas, where the entry/exit of vehicles and human operators is monitored through a gate, an economically viable increment in safety can be provided by means of removable and wearable sensors. This kind of monitoring architecture seems particularly useful

For understanding the figures in a better way, please refer to the online version of the article, available on the GEAM journal website.

Per una migliore comprensione delle figure, si rimanda alla versione online disponibile sul sito internet della rivista GEAM.

in logistic platform areas. A logistic platform is an area for the handling, storage and mobility of goods between different means of transport (e.g., planes, vessels, trains and trucks). This kind of platforms increases the efficiency of logistics flows to support productive areas. In addition to transport vehicles, several vehicles like reach stackers and personnel cars typically operate inside a logistic platform. Reach stackers, used to handle containers, can weight even 70 tons and reach a top speed of 25 Km/h with sudden acceleration. The presence of different kinds of vehicles poses serious safety problems for operators moving inside the logistic area. In this scenario, GNSS (Global Navigation Satellite Systems) tracking can help in mitigating the risk of collisions.

Tracking sensors based on GNSS navigation technologies were recently experienced in the Safeport FP7 project (Safeport project), that exploited the EGNOS (European Geostationary Navigation Overlay System), CDD (Commercial Data Distribution) and SoL (Safety of Life) services to achieve the high required accuracy (<10cm) to safely dock large vessels. GNSS technologies are also used in logistics to track shipments and storage of products (Mintsis *et al.*, 2004; Roula *et al.*, 2010).

The acronym GNSS refers to more than one constellation of satellites (mainly the American NAVSTAR GPS, the Russian GLONAS and the European Galileo) providing signals from space, transmitting positioning and timing data with a global coverage. A GNSS receiver determines its position by decoding such signals, with precision depending on the receiver type and on the elaboration technique (e.g., decametric/metric for navigational applications, centimetric and millimetric for technical and geodetic ones respectively; Groves, 2013; Hofmann-Wellenhof *et al.*, 2001; Leick *et al.*, 2015). Instead EGNOS (European Geostationary Navigation Overlay System) is the European satellite-based augmentation system (SBAS) that improves the accuracy and reliability of GNSS positioning by correcting signal measurement errors and by providing information about the integrity (i.e. the capacity to provide confidence thresholds as well as alarms in the event that anomalies occur in the positioning data) of its signals. It was deployed to provide safety of life navigation services to aviation, maritime and land-based users over most of Europe (ESA, 2006).

The NARVALO system is based on real time tracking of vehicles and humans by mean of built-in low-cost GNSS receivers, integrated with inertial sensors, called NARVALO boxes. Such receivers allow detecting position, speed, acceleration and direction of vehicles and humans with a centimetric precision. Then a Geographical Information System (GIS) collision avoidance algorithm, implemented by the team, predicts the risk of accident between vehicles and humans and a control center communicates alert or alarm to the mobile devices.

2. NARVALO *architecture*

The NARVALO architecture consists of three parts: a control center, a number of vehicle/human mobile devices and a remote cloud geoDB service. A wireless infrastructure is required for reliable communication between the control center and the mobile devices.

Each vehicle and human inside the monitored area is assigned to a mobile device and provided with a unique identifier used for its localization. The prototype of mobile devices (i.e., the NARVALO box) include a raspberry PI 3, a GNSS positioning sensor (ublox® NEO-M8T model), a low-cost inertial measurement unit (BerryIMU model), a wi-fi communication interface and an acoustic and visual signal. Each raspberry PI 3 is equipped with an installation of the Debian operating system and of the software for NRTK GNSS positioning based upon the open source library RTKLIB v. 2.4.3. The GNSS real time positioning is carried out in support of GNSS network positioning service. The GNSS receiver sends the raw data to RTKLIB software that at the same time receives the differential corrections and calculates the position. The accuracy and robustness of the NRTK solution is enhanced by inserting a new algorithm for detecting and removing multipath effects. The filtering algorithm, based upon a Multipath Detection Parameter (MDP), allows you to identify the observable suspected of suffering from multipath effects, and eliminates them in real time by the position calculation.

The interaction protocol between the control center and the mobile devices consists of the following steps. Each receiver continuously acquires information like positioning, direction, and speed of the monitored vehicle or human. These data are sent at a frequency higher than 1 second (i.e., between 1Hz and 5Hz) to the control center using a TCP/IP communication channel. Mobile devices can also receive alerts and alarms from the control center, which activate a visual or acoustic signal for the operator or on the vehicle.

The control center has a built-in server for handling the flow of data coming from the mobile devices. Data streams are redirected towards a real time collision avoidance module and a cloud service working as logging back-end for both the positioning data and the results computed by the collision avoidance algorithm. The interface of the logging system is defined via web services. The control center updates the current position of each vehicle or human in an ad-hoc data structure maintained in main memory and schedules the execution of the collision avoidance algorithm, that checks for possible collisions between humans and vehicles.

The remote cloud geoDB is used to maintain a complete log of the positions of humans and vehicles, the outcomes of the collision avoidance algorithm and the messages sent by the control center. The log can be used at any time to reconstruct trajectories of vehicles and humans. Availability and integrity of all the components and logged data can be ensured by maintaining replicas of stored data, and by an appropriate use of hardware and operating systems for real time data processing.

3. *The collision avoidance algorithm*

The NARVALO system is based on an innovative GIS collision avoidance algorithm, which runs in real time to detect possible risky situations which could lead to accidents, and consequently alerts the involved vehicles and humans. It is inspired to algorithms used in simulation of dynamic systems.

The input of the algorithm for each monitored vehicle and human are:

- the real time precise position, retrieved by using mass-market GNSS receiver and the NRTK GNSS positioning software, expressed as a set of coordinates (lat/lon) in the WGS84 Coordinate Reference System;
- velocity (module and direction) obtained from GNSS receiver;
- an integrity index coming from a multipath detection algorithm specifically developed within the NARVALO project and relying on the RTKLIB open source library;
- a standard deviation value obtained from the NRTK GNSS positioning software;
- only for vehicles, details about the shape of the vehicle (width and length) and relative positioning of the GNSS antenna on the vehicle both coming from cloud control center;
- only for vehicles, the data of acceleration, coming from a Inertial Measurement Unit (IMU) sensor, in the direction of the main axis x of a reference system which is inertial to the vehicle (Fig. 2).

The collision avoidance algorithm was developed using geospatial open source C libraries GDAL and OGR (GDAL and OGR library) to guarantee the real time requirements of the system.

The algorithm consists of three implemented functions (*Narvalo_Vehicle*, *Narvalo_Intersect*, *Narvalo_Collision*) and uses an approach with a preliminary approximation and a following refinement (Fig. 1).

For each pair of objects (i.e. monitored vehicles and humans), a coarse grain approximation of the positions that the two objects will assume in a given temporal windows (e.g., the next three seconds) are compared. This is achieved by constructing buffer polygons. The resulting polygons are then compared for intersection.

In order to reduce false positives, when an overlapping is detected we apply the following refinement step. An estimation of the precise positions of the objects in the next n (equi-distant) epochs is computed (e.g., instant +1, +2, +3 seconds with regards to the current instant/position). Then, for each instant, intersection of the resulting translated objects is checked. If a collision is detected, an alarm is sent to the operators associated to the objects (both humans and vehicles). In the following, additional details on the various functions executed by the algorithm are provided.

Narvalo_Vehicle computes the shape of the vehicle expressed in east and north coordinates in the Projected Coordinate Reference System of the logistic area (Fig. 2). It is not applied for humans since they are represented with a punctual geometry. It requires 8 input parameters: East and North coordinates, width and length of the vehicle, a and b parameters which determine the relative position of the GNSS antenna on the vehicle, azimuth of the velocity coming from GNSS and the acceleration in the *x* direction derived from IMU sensor.

Narvalo_Intersect relies on a coarse grain approximation of the positions positions of the objects. It compares the position of vehicles and humans in the logistic area using a variable buffer function in order to identify pairs of vehicle/human entities to be analyzed with more precision. In fact only the buffer which intersects with other buffers requires a refinement, while the buffers that have no intersection are not considered at risk and thus excluded by the last step of the algorithm (Fig. 3).

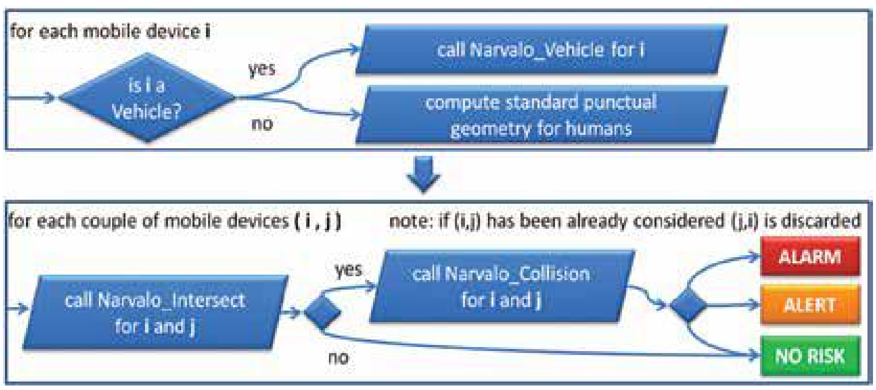

Fig. 1. Overview of the collision avoidance algorithm.
Panoramica dell'algoritmo di anti-collisione.

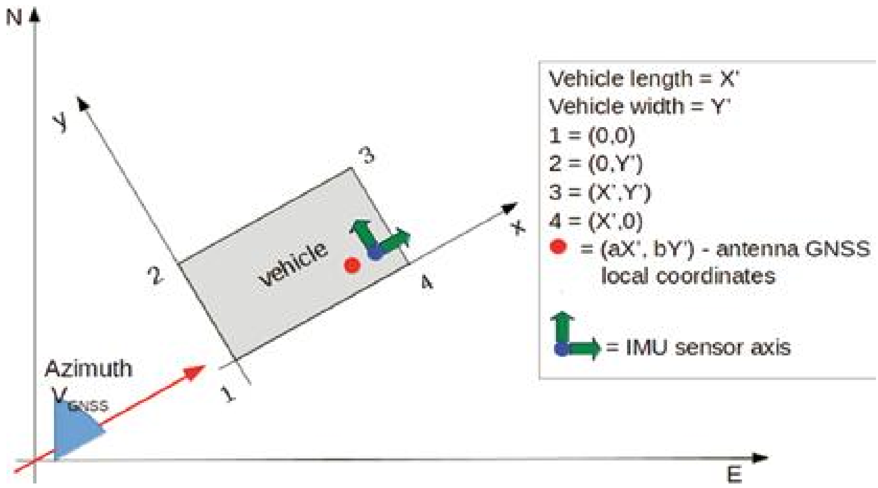

Fig. 2. Parameters and coordinate system defined in Narvalo_Vehicle.
Parametri e sistema di coordinate definiti in Narvalo_Vehicle.

Narvalo_Collision performs a refinement for the pairs identified by the *Narvalo_Intersect* function. It checks the real, precise expected positions of vehicles and humans in the next n (equi-distant) instants (e.g., instant +1, +2, +3 seconds w.r.t. the current instant/position), according to the module and direction of the speed derived from the GNSS data. If the expected positions do not intersect, the vehicles and/or humans are considered at *no risk*, otherwise they are considered at *risk* (case *Alarm* in Fig. 4).

If the route of a vehicle is not predictable, because the speed is null or too small, the output of the algorithm is an *Alert* for that vehicle and the nearby vehicles or humans previously individuated by the intersect function (case *Alert* in Fig. 4). Both intersect and collision functions do not take into account the collision between two humans.

4. The control center

The control center is a specialized middleware performing three main tasks: orchestrating the interactions with the mobile devices; scheduling and supporting the execution of the collision avoidance algorithm, and sending to the remote cloud-based geoDB all the data received from the mobile devices, as well as, the results computed by the collision avoidance algorithm.

The middleware is based on a specialized TCP/IP multithreaded server built using the C language. The server accepts requests on a predefined port. Upon receipt of a registration request, it creates a virtual copy of the requesting mobile device running as a separated thread inside the server. A new TCP/IP socket is confined in the newly created thread and exclusively used for handling the data streamed by the associated mobile device. Incoming messages to the control center can be of three different types:

- GNSS messages contain the precise positioning, speed, and direction of the mobile devices. The GNSS messages are sent by both kinds of mobile devices (i.e., for vehicles and humans).
- IMU messages contain the acceleration in the X and Y directions (Fig. 2) of the mobile devices. The IMU messages are available only for the vehicle mobile devices.
- Alarm messages notified to the control center by a human or by a vehicle operator, that foresees or observes a possible risk. In fact, NARVALO mobile devices are equipped with two buttons used to send alarms with two different levels of severity. The sent alarm message is then propagated to all the other mobile devices currently monitored by the NARVALO system.

The data contained in the three kinds of messages is stored in a pre-allocated data structure kept in main memory and shared between all threads communicating with the various mobile devices. The data structure kept in main memory maintains the most recent data received from each mobile device. We use split-locks (i.e. locks associated to each record of the data structure) to avoid the need of global synchronization of the entire data structure in case of concurrent access of multiple threads.

The collision avoidance algorithm is scheduled for execution every predefined amount of time

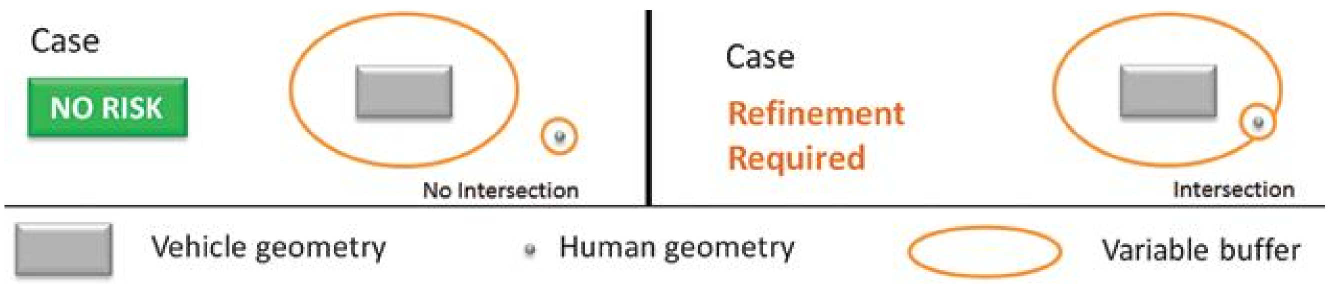

Fig. 3. Examples of results of Narvalo_Intersect.
Esempi di risultati di Narvalo_Intersect.

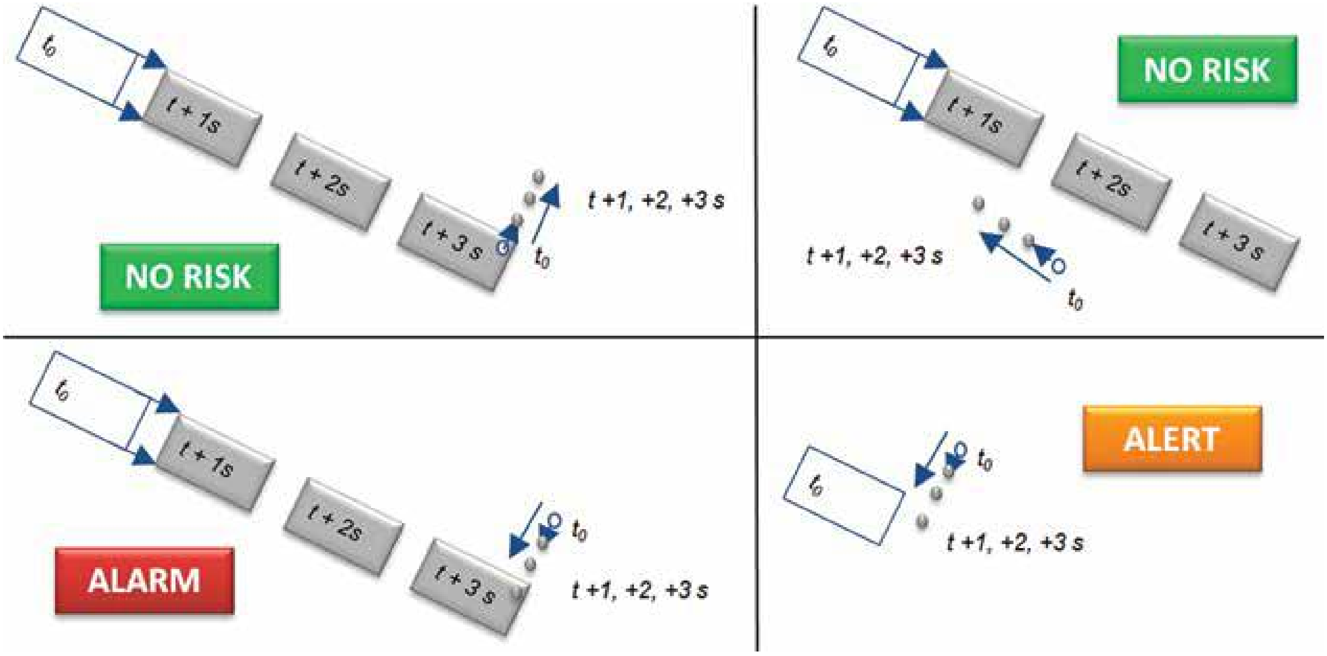

Fig. 4. Working scheme of Narvalo_Collision.
Schema di funzionamento di Narvalo_Collision.

(currently one second is considered sufficient). At each run, the control center creates a complete copy of the aforementioned data structure (the copy represents a snapshot of status of the monitored logistic area at the moment the algorithm starts its execution) and provides it as input to the collision avoidance algorithm. In this way, the various threads used to manage the incoming messages from the mobile devices can continue to save the information in the original data structure while, concurrently, the algorithm performs the required computations on the copy of the structure. Once the algorithm has finished its execution, it saves the results in the copied data structure.

At this point the control center sends a messages to each monitored mobile device by means of a dedicate thread for each device. Each message reports the results of the algorithm computed for the recipient mobile device. More in details, each message reports if an alarm or alert exists with every other monitored mobile device or if everything is fine (i.e. no risk). In this way mobile devices can be implemented ranging from simple acoustic alarm (for signaling if at least one alarm/alert has been received) to detailed graphical dashboards providing information about the relationship between the recipient mobile device and every other monitored mobile device (e.g., something like a radar view). Moreover, in case of a manual alarm raised by an operator, this is propagated to all the other mobile devices. Finally, the data are sent asynchronously to a cloud-based logging system (geoDB). The cloud-based logging system can be accessed remotely via web services. The services are invoked using HTTP connections. Data sent to the logging systems are stored in a geoDB to ease the rendering phase of the monitored data (e.g. via maps, marks associated to "humans" and vehicles, and trajectories extracted from historical data).

5. Tests and future perspectives

Several in-house tests of the NARVALO system were performed during the last months. Focusing on the ones regarding the algorithm and the control center, they were very useful for debugging and verifying that both modules adhere to the user and systems requirements. Several made-up routes simulating alert, alarm and no risk situations, sent by ad-hoc virtual mobile devices installed on the real NARVALO boxes, were used to debug and refine the implementation of the algorithm and the control center.

At the end of the development process the combination of algorithm and control center performs as expected in all the tested cases simulating with virtual mobile devices situations of alarm, alert and no risk. Moreover, in our test scenarios, composed by five vehicle or humans devices, the algorithm was able to complete all the computation in just a few milliseconds in the worst cases (i.e., when all vehicles/humans requires the refinement step of the collision avoidance algorithm). This is by far lower than the maximum required computational time and might allow the adoption of the NARVALO system also in moderately crowded contexts.

Finally, also the communications from/to the virtual mobile clients and the remote cloud server perform as expected.

As future work we plan to better analyze the scalability of the algorithm with regard to the number of monitored mobile devices. Moreover, soon operational demonstration of the prototype system will be performed at the logistic area of the main stakeholder of NARVALO project, i.e. Inter Repair North in Livorno.

References

Christer, E., 2004. *Real-Time Collision Detection*. CRC Press, Inc., Boca Raton, FL, USA.

Deng-Yuan, H., Chao-Ho, C., Tsong-Yi, C., Wu-Chih, H.uc, Kai-Wei, F., 2017. *Vehicle Detection and Inter-vehicle Distance Estimation Using Single-lens Video Camera on Urban/Suburb Roads.* Journal of Visual Communication and Image Representation, In Press. DOI: 10.1016/j.jvcir.2017.04.006

EGNOS system: https://www.gsa.europa.eu/egnos/what-egnos – accessed on 07/05/2017

ESA, 2006. *EGNOS – The European Geostationary Navigation Overlay System – A Cornerstone of Galileo*. Publication: The European EGNOS Project (ESA SP-1303), Editors: Bruce Battrick & Dorothea Danesy, ISBN: 92-9092-453-5.

GDAL and OGR library: http://www.gdal.org/ – accessed on 07/05/2017

Groves, P.D., 2013. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*, Second Edition, Artech House, 776 pp, ISBN-13: 978-1-60807-005-3

Hofmann-Wellenhof, B., Lichtenegger, H., Collins, J., 2001. *Global Positioning System – Theory and Practice*. Springer&Verlag, Vienna, 382 pp, e-ISBN: 978-3-7091-6199-9.

Kolk, H., Kirschbichler, S., Tomasch, E., Hoschopf, H., Luttenberger, P., Sinz, W., 2016. *Prospective Evaluation of the Collision Severity L7e Vehicles Considering a Collision Mitigation System.* Transportation Research Procedia, Volume 14, 2016, Pages 3877-3885. DOI: 10.1016/j.trpro.2016.05.474

Leick, A., Rapoport, L., Tatarnikov, D., 2015. *GPS Satellite Surveying*, 4th Edition, John Wiley & Sons Inc., New Jersey, USA, ISBN: 978-1-118-67557-1.

Mintsis, G., Basbas, S., Papaioannou, P., Taxiltaris, C. and Tziavos, I.N. 2004. *Applications of GPS technology in the land transportation system*, European Journal of Operational Research, Volume 152, Issue 2, 16 January 2004, pp. 399-409.

RTKLIB Open Source Software for GNSS Positioning: http://www.rtklib.com/ – accessed on 07/05/2017

Roula, M., Zenon, M., Dimitrios, N., 2010. *Optimisation of Logistics Operations Using Gps Technology Solutions: A Case Study*. Poms – 21st Annual Conference Vancouver, Canada, pp. 1-17, https://www.pomsmeetings.org/confpapers/015/015-0822.pdf

Safeport project: http://cordis.europa.eu/project/rcn/206623_it.html – accessed on 07/05/2017

Xueming, W., Jinhui, T., Jianwei, N., Xiaoke, Z., 2016. *Vision-based two-step brake detection method for vehicle collision avoidance*. Neurocomputing, Volume 173, Part 2, Pages 450-461. DOI: 10.1016/j.neucom.2015.04.117.

Acknowledgments

The NARVALO project is co-funded by ASI, Agenzia Spaziale Italiana. The authors thank ASI, the project partners 'Optisoft srl' and 'Gter srl Innovazione in Geomatica, GNSS e GIS', and the stakeholder Inter Repair Nord that will host the operational demonstration of the prototype system.