Real-Time Path Planning and Obstacle Avoidance for RAIS: An Autonomous Underwater Vehicle

Gianluca Antonelli, Stefano Chiaverini, Roberto Finotello, and Riccardo Schiavon

***Abstract*—This paper describes a navigation and guidance system (NGS) with real-time path planning and obstacle avoidance capabilities that has been developed for the autonomous underwater vehicle RAIS. The vehicle is designed to accomplish two missions: pre-deployment survey of sea bottom, and visual inspection of pipelines. In the first mission, the NGS must be able to track a predefined path while avoiding the unplanned occurrence of obstacles. In the second mission, the NGS must track a pipeline by locally reconstructing its location from visual information; also in this case, the unplanned occurrence of obstacles must be handled. Furthermore, the NGS must properly take into account the presence of ocean current and some drastic constraints due to sensor and actuator characteristics. Numerical and hardware-in-the-loop simulations have been developed to verify the effectiveness of the proposed NGS.**

***Index Terms*—Autonomous underwater vehicles, navigation and guidance systems, obstacle avoidance, path planning.**

I. Introduction

AUTONOMOUS underwater vehicles (AUVs) are becoming of increasing importance to accomplish underwater tasks, e.g., sea bottom exploration or pipeline/cable inspection, since they reduce costs and avoid the risks of systems requiring a human operator on board. Nevertheless, AUVs suffer from evident power, sensing, and communication limitations, thus it is crucial to develop a robust and safe on-board control system and built-in machine intelligence. Reference [21] presents a summary of several existing AUVs and describes their main characteristics. A remotely operated vehicle is described in [18] which has been simulated in missions involving visual feedback of pipes.

A major challenge in the development of advanced autonomous systems is the realization of a real-time path planning and obstacle avoidance strategy which can effectively navigate and guide the vehicle in unstructured environments. In the technical literature, the terms navigation and guidance have different meanings. In fact, the term *navigation* is mainly used with reference to the task of building a map of the environment and to the localization of autonomous vehicles. Recently, it has been used also with reference to higher level tasks, such as *contextualization* of the vehicle in the environment [9], [19]. On the other hand, *guidance* is intended as the task of generating the reference trajectory (heading, speed and pitch for AUVs) to the low-level dynamic controller [11]. In some cases there is an overlap between the operation of the navigation system, and the guidance systems, e.g., when implementing a reactive behavior of the vehicle. For this reason, in the following, the two terms will be used together under the expression *Navigation and Guidance System* (NGS).

While there is a wide literature about NGSs for land autonomous vehicles (see e.g., [7], [15], [16]), a relatively little number of papers report NGSs specifically designed for underwater applications. In addition, the specific characteristics of AUVs do not allow a straightforward extension of most motion algorithms designed for land vehicles to underwater vehicles.

Among the papers describing NGSs for AUVs, reference [22] presents a motion planning algorithm based on an annotated map; the path planner retrieves an old, matching, route from the database and modifies it to suit the new situation. This method requires *a priori* information on the environment and off-line computations. Reference [12] proposes an on-line 3-D exploring algorithm which generates a mosaicked image of the ocean floor. Reference [17] presents a hybrid (i.e., discrete/continuous state) controller for AUVs. The controller is composed of three levels: a first, discrete, *strategic level* that labels the current vehicle state and the two, continuous, *execution level* and *tactical level*. These implement low-level functions that are respectively synchronous and asynchronous with respect to time. Reference [8] focuses on an NGS developed for cable inspection. In [24], an elliptical virtual force field (VFF) is used to command a bottom-following behavior to the Bentic Explorer. In that case, because of the specific mission of the vehicle, obstacle avoidance is pursued only in the vertical plane. In [23], a potential-field based navigator is presented that iteratively modifies the path starting from a first *global* tentative one. This approach, however, does not avoid the risk of being trapped in local minima of the potential field. Reference [13] presents an interesting mine avoidance technique for AUVs. Based on a three-dimensional discretization of the environment, the path planning technique consists of computing a safe path avoiding the unsafe cells of the map. When some conditions occur, the vehicle has to make a 360° turn to map the environment close to it and then generates a safe path. The simulations reported in the paper are realistic and take into account the vehicle dynamics and the sensors characteristics.

This paper presents an NGS developed for an AUV named RAIS, which is the outcome of a collaborative research program managed by Centro Oceanologico Mediterraneo (CEOM) and

Manuscript received August 1, 2000; revised January 29, 2001. This work was supported by the Italian R&D Directorate and ENI.

G. Antonelli and S. Chiaverini are with the Dipartimento di Automazione, Elettromagnetismo, Ingegneria dell'Informazione e Matematica Industriale, Università degli Studi di Cassino, 03043 Cassino (FR), Italy (e-mail: antonelli@unicas.it; chiaverini@unicas.it).

R. Finotello and R. Schiavon are with Tecnomare S.p.A., 30124, Venice, Italy (e-mail: finotello.r@tecnomare.it; schiavon.r@tecnomare.it).

Publisher Item Identifier S 0364-9059(01)03694-9.

developed with the technical contribution of Tecnomare, Snamprogetti, and SASP Offshore Engineering. A preliminary description of the NGS can be found in [2], [3]. The NGS of RAIS must allow accomplishment of two different missions: inspection of a pipeline and pre-deployment survey of the sea bottom. In addition, proper operation of the NGS must be ensured under the following design constraints:

- a suitable constant speed value of the vehicle must be ensured, since the performance of the sensory system is strongly affected by the cruise speed;
- a feasible motion must be commanded to the low-level control of the vehicle; in particular, the motion trajectory should accommodate the nonholonomic constraints deriving from the actuating system of the vehicle;
- only few major characteristics of the environment are available at the mission planning level (e.g., maximum curvature of the sea bottom in the vertical plane, maximum slope of the sea bottom, maximum curvature of the pipeline in the horizontal plane), while the presence and position of the obstacles as well as the exact pipeline posture and the bottom shape are unknown;
- the effects of a linear ocean current, whose strength and direction is not known in advance, must be counteracted.

The proposed NGS has first been tested by extensive computer simulation runs aimed at verifying the fulfillment of the missions' requirements and analyzing its performance in a number of different cases. Then, it has been implemented in the vehicle controller and integrated with the man machine interface (MMI) dedicated to the system programming and monitoring. The implemented NGS has finally undergone hardware-in-the-loop tests, i.e., with a real-time digital computer that emulates the dynamics of the vehicle in a simulated underwater environment.

II. Vehicle Description

The preliminary design of the vehicle has been completed and two views of RAIS are reported in Fig. 1.

The vehicle is torpedo-like shaped and is composed of three main parts: the bow, the stern and the vessel body. The vessel body is the only water-sealed part and contains the electronic components. The power supply system, based on secondary batteries, is installed in the lower part of the vehicle. The fore and aft are water filled and contain the instrumentation and the thrusters. The communication with the surface control station, only for emergency purposes, is guaranteed by an acoustic link.

The vehicle has to operate up to 1000-m-depth, relaxed to 400 m for the first prototype. Key features of the prototype vehicle are summarized in Table I, while the main vehicle subsystems are briefly described in the following.

- **Vision and Automatic Recognition subsystem**—It carries out the analysis of the scene the AUV is approaching. In detail, its main task is to identify the pipe border lines and the sacrificial anodes. This subsystem is based on neural network algorithms [10] along with statistics and morphological analysis of the images. Moreover, the vision system is in charge of getting a three–dimensional (3-D) sealine profile. This is achieved by using two "whiskers" pinned at the AUV hull sides, each carrying a green YAG laser to provide structured light. Tests based on original TV images have been carried out and show the reliability of the implemented techniques [10]. The outputs of such processing are then fed to the positioning system, allowing the searching and tracking of the pipe and the accurate identification of the location of the sacrificial anodes.
- **Position and Speed Measurement subsystem**—The positioning system is based on the integration of information coming from different data acquisition systems. To carry out such integration an extended Kalman filter is used. This allows to mix different measurements (such as speed and position), providing the best estimate of the vehicle overall state and taking into account the errors affecting the available sensors. The sensory system includes the following devices: Doppler sonar for the speed measurements; fiber optic gyro for the attitude

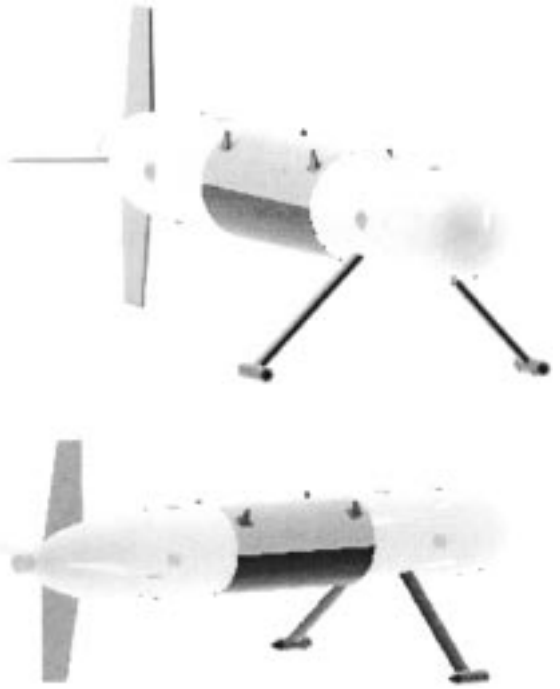

Fig. 1. Preliminary design of the RAIS vehicle. It has a common torpedo-like shape; the two "whiskers" used to implement the visual feedback can be recognized.

TABLE I
RAIS Data

Length	5.50 [m]
Vessel Diameter	0.75 [m]
Overall Clearance	2.10 [m]
Weight	1000 [kg]
Maximum Speed	2 [m/s]
Maximum Depth	400 [m]
Battery Perform	8 [h]
Thruster Power	300 [N]
Transversal Water Current Allowance	0.5 [m/s]
Vessel Material	Aluminum Alloy
Hull Material	Carbonium & Kevlar Compound

information; pressure sensor for the depth measurements; echosounder and acoustic positioning for the sea bottom data and a panoramic sonar for the obstacle detection. The sealine position and attitude with respect to the vehicle is evaluated by integrating the data coming from the Vision system, which provides the pipe direction, with those obtained via laser triangulation. The laboratory tests have shown an accuracy in the order of some millimeter at 2-m distance.

- **Thrust and Maneuvering subsystem**—It is based on a thruster parallel to the fore–aft direction, which is used to control the surge velocity, and on two control surfaces (rudder and elevator) set in the vehicle aft, which are used to control vehicle's yaw and pitch at high velocity. Moreover, the vehicle is equipped with four small thrusters to be used in hovering manoeuvres at small velocities, i.e., when the control surfaces are not usable.
- **Energy storage subsystem**—The choice is limited to secondary rechargeable cells to maintain the costs at a reasonable level. Silver–zinc cells have been chosen since they comply the safety requirements about the risks related to the hydrogen emission, while they can supply a considerable amount of energy (and power) and can operate at high temperatures without failures. Moreover, the voltage of the cells is slightly influenced by the charge state, the temperature and the load conditions. The main drawback of these cells is that the recharge time is considerably longer than other available technologies.
- **Navigation and Mission Control subsystem**—It relies on the information coming from the position and speed measurement subsystem to on-line determine the trajectory the AUV has to follow. The trajectory is planned in order to track the sealine and the sea bottom as accurately as possible and to avoid obstacles. This is the main subject of this paper; thus, a detailed description of issues related to this subsystem is reported in the next Sections.

III. Missions Description

The two missions to be accomplished are:

- **Mission 1**—High resolution pre-deployment survey of sea bottom and sub-bottom.
- **Mission 2**—Sealine detailed visual inspection and free span gauging.

The planning and construction of a submarine pipeline requires a complete characterization of both bathymetry and morphology of the sea floor and sedimentological characteristics of the first layers under the bottom. The main limits of the current technology are bounded to deep-water surveys, more than 1000 m, where the precision reachable with equipment installed on a vessel is not sufficient for engineering purposes. Even the new development of bathymetric systems towed to a towfish or installed on a remotely operated vehicle (ROV) cannot solve completely the problem, due to low precision of the positioning of the towfish in deep waters, and in case of ROV installation, due to need for an expensive vessel support.

In case of Mission 2, the ROV current technology permits to obtain satisfactory results; the main problem connected to this kind of survey is represented by the high cost due to the use of a large team of technicians and high performance ships, particularly for deep water, where a bulky and heavy umbilical winch is needed. The aim of the mission is to collect data about the position and the surface status of the sealine and, if necessary, to measure free span characteristics.

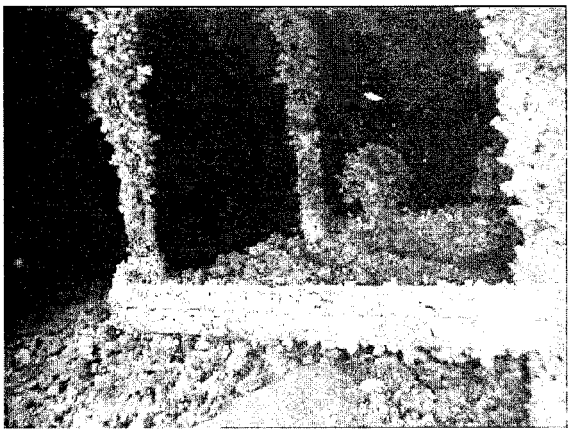

Fig. 2. Pipeline mechanical support (example). This structure is used to anchor the pipeline at the sea bottom.

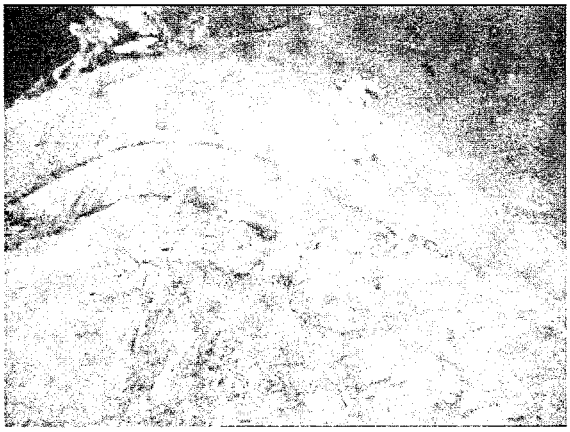

Fig. 3. Sealine mattress (example). It is used to hold the pipeline at the sea bottom.

For both missions the technical requirements are:

- Constant speed [$1 \div 2$ m/s] cruise of the vehicle.
- Unknown obstacles must be detected and the trajectory must be locally modified in order to avoid them.
- The vehicle has to cruise in presence of water current up to 0.5 m/s with any direction with respect to it.
- The minimum curvature radius of the sea bottom is 5000 m and its maximum slope is 10%.
- The vehicle must be able to cruise at constant distance from the sea-bottom/pipeline or at constant depth.

In Mission 1, the path is given in terms of a pre-programmed path defined in absolute coordinates. In Mission 2, the vehicle must reconstruct in real-time the pipeline position from visual feedback; it is assumed that the pipeline has a minimum curvature radius on the horizontal plane of 5000 m.

Some obstacles to be avoided are typical objects due to sealine installation such as mechanical supports (Fig. 2) and mattresses (Fig. 3).

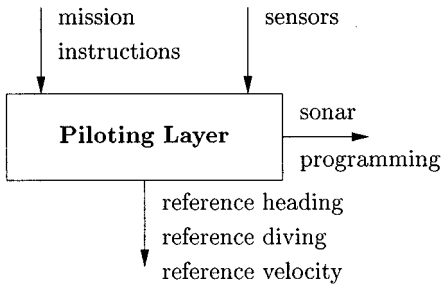

Fig. 4. Input–output block representation of the PL.

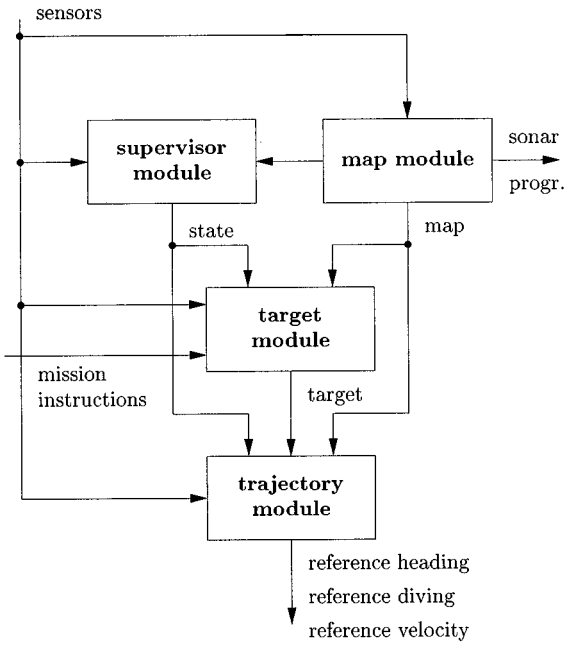

Fig. 5. PL functional architecture.

IV. Navigation and Guidance System

The NGS consists of a piloting layer (PL) that generates the reference trajectory of the vehicle in real time. An overall block representation of the PL is reported in Fig. 4; its inputs are the sensor's readings and the high-level mission instructions, whereas its outputs are the heading, diving and velocity references and some programming instructions to the panoramic sonar. An exploded view of the functional architecture of the PL is instead given in Fig. 5, where the main functional modules and the relevant data flow are evidenced. A description of the single functional modules is provided later on in this Section.

Due to the large dimension of the environment and to the uncertainty affecting its description, only a local approach to the trajectory generation problem is possible. However, it is well known that local motion planning techniques can cause the vehicle being trapped at some point or in a closed path; therefore, the local strategy has been integrated in a hierarchical structure that has the task to detect the occurrence of traps and to activate suitable escape strategies.

In view of the characteristics of actuators and sensors, the motion trajectory generation is performed independently on the horizontal and vertical plane, respectively. In fact, the vehicle is designed in order to be stable in the roll degree of motion.

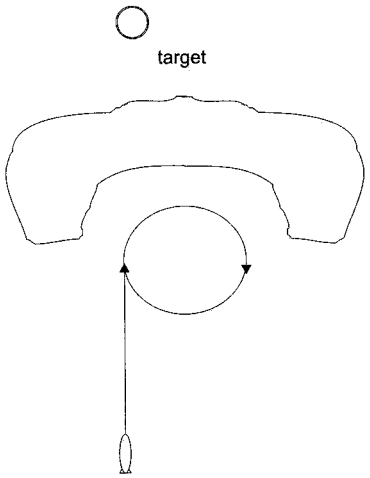

Fig. 6. Example of a trap-like situation. VFF-based techniques cannot guarantee avoidance of such situations.

In the horizontal plane the PL strategy is based on concepts from the VFF approach [4] and on some geometric considerations made in [13]. However, since the VFF approach is not adequate—and can even lead to unstable behavior of the vehicle—in presence of an obstacle in the forward direction or in a narrow corridor [14], it has been integrated by a geometrical approach which increases the vehicle reactivity and avoids the above problems. This has been done to consider also *polar* information instead of only cartesian one as in the VFF; similar considerations have motivated the work in [5], [6] and, more recently, in [20].

When planning the motion trajectory on the horizontal plane, the PL also has the task to detect local minima which constitute traps for potential-based approaches (see e.g., [15]). A typical local minimum for VFF techniques is the presence of a **U** shaped obstacle; in this case the vehicle could be driven by the algorithm to an indefinite turn in a round (see Fig. 6). Another specific dangerous situation to be detected is the occurrence of a *canyon*, i.e., an obstacle with a large entrance that becomes narrow in the end. In fact, due to the minimum turn radius constraint of the vehicle, if the vehicle enters the canyon a safe path is not guaranteed anymore (see Fig. 7).

In the vertical plane the PL adopts a simple VFF strategy since trap situations do not occur.

Based on the above strategies, the vehicle is driven by means of a target point whose motion describes the reference trajectory. At a given time instant, the current target is a virtual point that pulls the vehicle on the desired path (the pipeline or the planned bottom survey path). In Mission 2 the target is generated based on visual information; when the vehicle is far from the pipeline, instead, the target is generated based on a real-time estimation of the pipeline position and direction.

The thruster must be commanded in a way to generate constant velocity along the advancing direction; this is required to allow proper operation of the sensory system.

The proposed solution must satisfy the following design constraints:

- real-time operation;
- limited computational load;
- robustness;
- modularity.

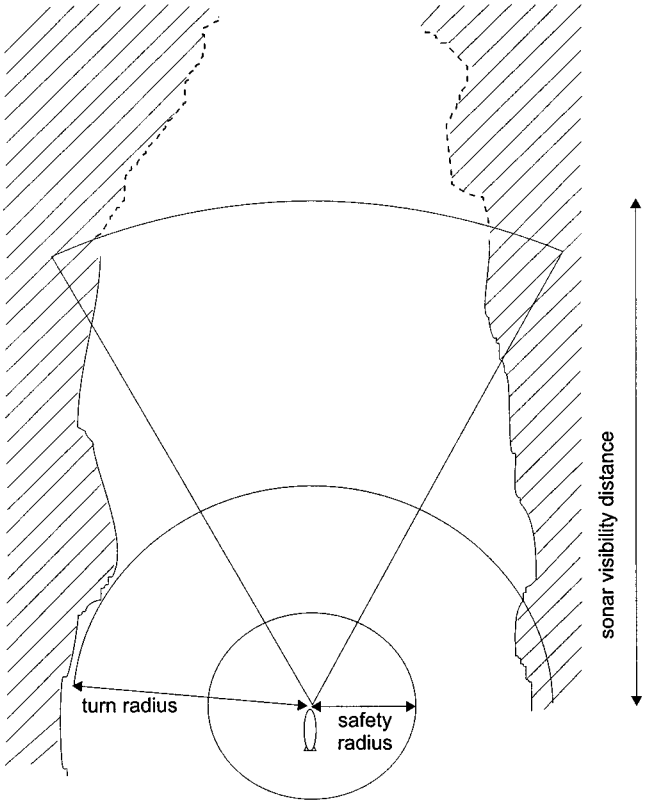

Fig. 7. Example of a canyon-like environment. This situation is critical for vehicles with a minimum turn radius. In fact, if the canyon gets narrower than the minimum space necessary to make a 180° turn, the vehicle cannot generate a safe path if the advancing direction is obstructed.

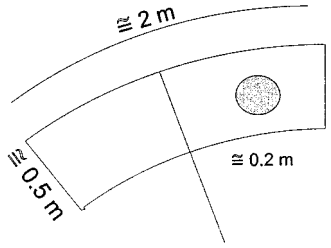

Fig. 8. At the maximum distance (50 m), an object of 20 cm is detected by the panoramic sonar with an uncertainty of 0.5×2 m.

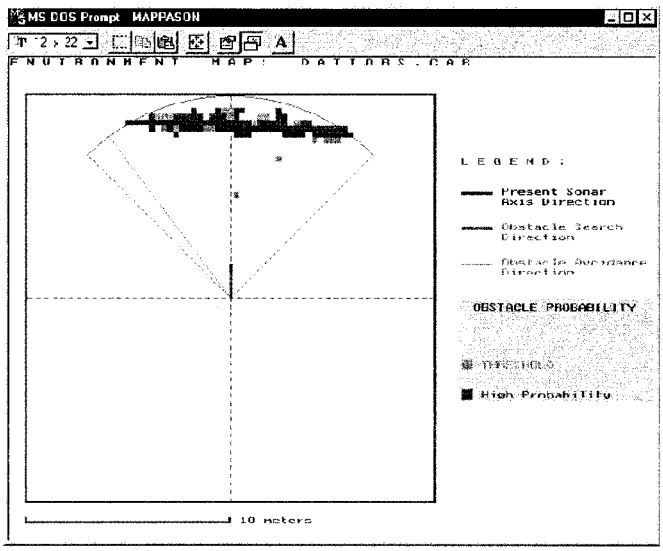

Fig. 9. Example of a map for a scaled version of the sonar: wall in front of the sonar in a pool.

A. Map Module

The *map* module has the task to update the environmental information concerning obstacles' position. Due to the huge path dimension, the map is local and the information about obstacles' position is deleted when the vehicle goes far from them; in case of sea bottom survey the environmental information is stored together with the vehicle position to allow off-line data processing. The output of the module is continuously updated in vehicle-fixed frame; when a new sonar reading is ready—typically at a lower frequency than the path generation rate—the new data are integrated with those stored in the former map. It must be noted that the vehicle is designed in order to be used in an environment whose characteristic are partially known: the sea between Tunisia and Sicily (Italy), where a flat bottom with localized clusters of stones is expected. This partial *a priori* knowledge has been taken into consideration in the choice of the sensor characteristics and in the development of the map module.

The map module takes into account the characteristics of the panoramic sonar:

- It spans a specified angular sector that can be modified in real time by the PL. The data are available only after a complete angular sector reading, this means 2 s for an 18° sector with a step of 1.8° or 10 s for a 90° sector with the same step. The vehicle position associated to each sonar reading has to be taken into account in order to update the data in the vehicle-fixed frame;
- The size of the vector reading is function of the angular sector size and the step size (10 and 50 in the previous cases);
- The cone of each sonar reading is 2.3° and its range is 50 m; at 50 m the corresponding arc is about 2 m. This implies that an obstacle of ≈20 cm is seen with ≈1 m of uncertainty. Moreover, the accuracy of the sonar on the distance measurement must be taken into account (see Fig. 8). The resolution of the map, thus, is also function of the distance of the obstacle when it has been detected.

A scaled version of the sonar that will be mounted on RAIS has been tested in a pool. In Fig. 9, the result of a test of the sonar in front of a vertical wall is presented [1].

It is then possible to summarize the map module working cycle in the following steps:

1) update the former map, expressed in the vehicle-fixed coordinates of the previous position, in the current vehicle-fixed coordinates;
2) eliminate from the current map the data no longer useful to plan the path; these are stored for off-line processing in the case of a sea bottom survey mission;
3) acquire the new data when available.

Some comments on accuracy of the map are finally of concern.

Since the vehicle position is obtained by integration of velocity measurements, it is affected by a drift error that grows linearly with the distance traveled from the beginning of the mission. The importance of this error depends on different considerations; for instance, in case of sea bottom exploration, macroscopic information is sufficient to localize a proper site where

the pipeline can be deployed and punctual information might not be necessary. In any case, two method to reset the error on absolute position of the vehicle have been considered; namely, the common Global Positioning System (GPS) reset method (i.e., the vehicle comes to the surface and resets its position with a GPS sensor), and a reset based on the detection of known landmarks such as sacrificial anodes via the Vision system. The latter method is under study for application to the case of Mission 2.

On the other hand, to the purpose of obstacle avoidance, only the error affecting the relative position between vehicle and obstacle has to be considered. To avoid an obstacle without risks, thus, it is sufficient to consider a conservative path that guarantees a distance from the obstacle larger than this error. In our case, since the panoramic sonar has a range of 50 m the error on the relative vehicle-to-obstacle distance is negligible.

B. Supervisor Module

The *supervisor* module detects the occurrence of traps or situations that are potentially dangerous for a cruising vehicle that advances at a constant velocity. Based on the map and the sensor information, the supervisor detects a *state* of the vehicle; at each state a different trajectory generation strategy is used. For example, if the supervisor detects a local minimum it switches to a state that activates an escape strategy; also, the target generation is different when the vehicle is exactly on the pipeline or far from it. The output of the module, thus, is the state of the vehicle, i.e., a label to be used from the target module and the trajectory generation module.

Let first give some definition that will be useful later:

- *safety radius*—Distance from the origin of the vehicle-fixed frame below which no obstacles have to be detected at any moment of the mission; this is also the distance needed by the vehicle to stop. Since the vehicle must advance with constant velocity, the PL must always preserve the integrity of a *safety circle* of radius equal to the safety radius.
- *turn radius*—Distance from the origin of the vehicle-fixed frame which characterizes a *turn circle* within which the vehicle is able to make a turn of 180° without modifying its velocity and depth. The turn radius is twice the minimum radius of curvature that can be achieved by the vehicle (see Fig. 10).
- *turn halfcircle*—Dividing the turn circle by the diameter parallel to the advancing direction gives two halfcircles. Keeping *one* of these two halfcircles always free from obstacles allows to obtain a safe behavior of the vehicle avoiding it entering in canyon-shaped obstacles. Fig. 7 illustrates a canyon-like environment with large entrance and narrow end; it can be easily recognized that entering this type of canyon is dangerous for the vehicle's safety since it may be not possible to generate a safe path. This situation is avoided using the two turn halfcircles as explained later.

In the following, the states of the vehicle to be detected by the supervisor are defined:

- `Trap`—VFF based algorithms, and in general local path planning algorithms, can cause paths as the one shown in Fig. 6, where the vehicle makes an indefinite turn around a circle. In the literature several techniques to detect such situation are described. The escape procedure adopted in our supervisor is based on the wall following method (WFM) (see, e.g., [4]).
- `Canyon`—As already shown in Fig. 7, a canyon shaped obstacle can be critical for vehicle with nonholonomic motion and constant advancing velocity constraints. The PL avoids entering in canyons just monitoring the two turn halfcircles: as soon as they both are violated it generates a turn in the direction of the last violated halfcircle thus restoring its integrity.
- `Not_visible`—The vehicle is in the `Not_visible` state if the other states are not activated and the pipeline is not visible, i.e., is outside the scene seen by the Vision system. This state, of course, does not apply for Mission 1. The trajectory is generated via a VFF algorithm, which computes the current target based on an on-line estimation of the pipeline position and orientation.
- `Nominal`—The vehicle is in the Nominal state if the other states are not activated and the pipeline is visible. In this state, the trajectory is generated via a VFF algorithm.

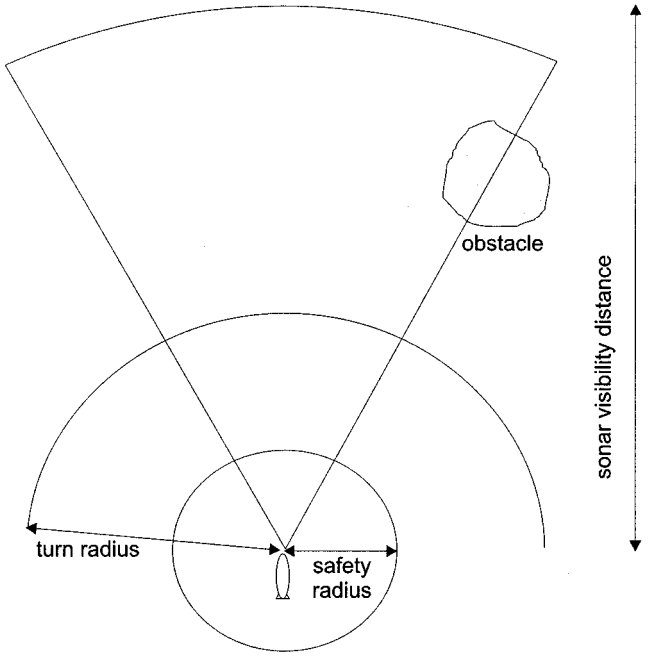

Fig. 10. Definition of the safety and turn radius on the horizontal plane. The turn radius is the double of the curvature radius. If the turn circle is free from obstacles, the vehicle can make a 180° turn.

The supervisor must properly detect the current vehicle state and switch from one state to the other. It has been implemented with a finite state system for which the transition is activated by rules.

As mentioned before, the VFF algorithm has been integrated with a geometrical technique that is in charge of handling the situations which are not properly handled by the VFF technique [14]: narrow corridors, labyrinth-like environments, and the presence of an obstacle in the forward direction. The first two situations are avoided by the use of the state `Canyon`. The presence of an obstacle in the forward direction is handled by modifying the definition of the turn halfcircle. In detail, we define the two halfcircles in a way that they overlap of a certain angular sector across the forward direction; therefore, when the vehicle approaches an obstacle it violates both halfcircles at the same time and the vehicle

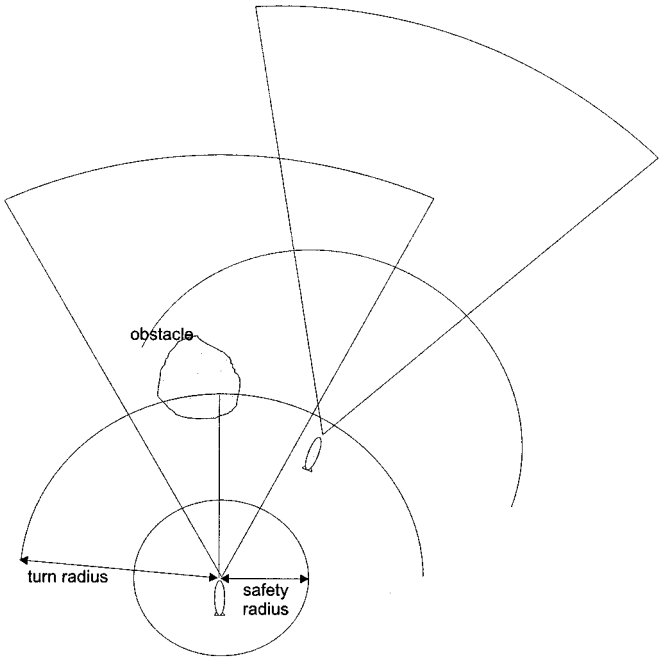

Fig. 11. A suitable definition of the turn halfcircles gives the vehicle a reactive behavior in case of obstacles in the forward direction. The figure shows the vehicle in the situation where the obstacle violates both the halfcircles and the movement made to avoid it.

will turn until the integrity of one halfcircle is preserved, i.e., until the obstacle is no longer in the forward direction. Fig. 11 shows the vehicle before and after avoiding the obstacle. Notice that this geometrical technique gives the vehicle a safe and reactive behavior taking into account its motion constraints.

C. Target Module

The *target* module generates the current target to be reached by the vehicle so as to follow the nominal path as close as possible while avoiding obstacles. Notice that the target is not constant during the mission execution. The inputs of the module are the sensor readings, the state of the vehicle, the map data, and the mission instructions; the output is the current target in vehicle-fixed frame.

The required nominal path for Mission 1 is typically composed by straight line segments, while for Mission 2 is the axis of the pipeline to be inspected. In the latter case, a geometric description of the path is not available. This is due to two main reasons: a geometric representation of the pipeline would not be accurate enough; the pipeline is subjected to small translation caused by the ocean current. Therefore, the nominal path for Mission 2 is reconstructed on-line from sensory data.

In any case, the target generation is mainly based on the information provided by the Vision system. If, during an obstacle avoidance maneuver, the vehicle goes far from the pipeline, the current target must be updated in a suitable way so as to drive the vehicle over the pipeline again. An estimation of the pipeline position is used to update the target. Due to the errors affecting the estimation of the pipeline's direction made by the Vision system, and the eventual change in the pipeline's curvature, the confidence in this estimation decreases with the distance traveled far from the pipeline. Eventually this information can activate an emergency.

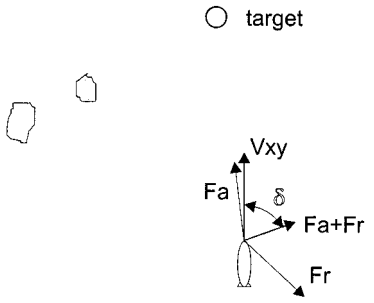

Fig. 12. VFF in the horizontal plane.

D. Trajectory Module

Depending on the state of the vehicle, the *trajectory* module implements the trajectory generation strategy.

The thruster is commanded in a way to guarantee that the advancing velocity, i.e., the inertial linear velocity, is kept constant all over the mission. Notice that the advancing direction may not be parallel to the thruster action due to the presence of ocean current.

The VFF technique implemented on the horizontal plane is used when the supervisor detects the states `Nominal` and `Not_visible`. Basically there might be obstacles in the current map but they do not cause the activation of the states `Trap` or `Canyon`. The algorithm implemented combines an attractive force $\boldsymbol{F}_a$ toward the current target and a repulsive force $\boldsymbol{F}_r$ from the obstacles [4]. If we define as $\hat{\boldsymbol{r}}_t$ the unit vector characterizing the direction from the vehicle to the target, and as $\boldsymbol{r}_i$ the position of the obstacles in vehicle-fixed frame we can compute the virtual force $\boldsymbol{F}$ acting on the vehicle as

$$\boldsymbol{F} = \boldsymbol{F}_a + \boldsymbol{F}_r = k_a \hat{\boldsymbol{r}}_t - k_r \sum_i \frac{\hat{\boldsymbol{r}}_i}{||\boldsymbol{r}_i||^2} \tag{1}$$

where k_a and k_r are design gains. To preserve the safety circle and continuity with respect to the limited visibility of the sonar, eq. (1) can be modified as

$$\boldsymbol{F} = \boldsymbol{F}_a + \boldsymbol{F}_r = k_a \hat{\boldsymbol{r}}_t - k_r \sum_i \hat{\boldsymbol{r}}_i \left[\frac{1}{(||\boldsymbol{r}_i|| - r_s)^2} - \frac{1}{(r_{sen} - r_s)^2} \right] \tag{2}$$

where r_s is the safety radius and r_{sen} is the visibility distance of the sonar. Notice that this guarantees that the module of the repulsive force becomes infinity at a distance r_s from the vehicle, thus preserving a safety circle around it. The second term is aimed at having null repulsive contribution for the obstacles at the boundaries of the visibility circle.

The angle δ between the advancing direction and the direction of the virtual force acting on the vehicle is used to modify the yaw rate as

$$\dot{\psi} = k_s \delta \qquad \delta \in [-\pi, \pi] \tag{3}$$

with k_s being a design gain. Fig. 12 reports a sketch of the vehicle under the action of the VFF algorithm in the horizontal plane.

When the supervisor detects the state `Canyon` the VFF algorithm is disabled and a turn is commanded with constant yaw rate. Notice that, due to the integrity of one of the turn halfcircles, this command is executed in safe conditions.

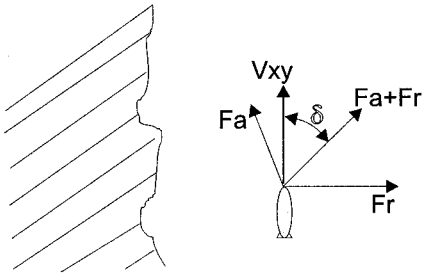

Fig. 13. Following method (horizontal view). The attractive force, in this case, depends on the repulsive force direction.

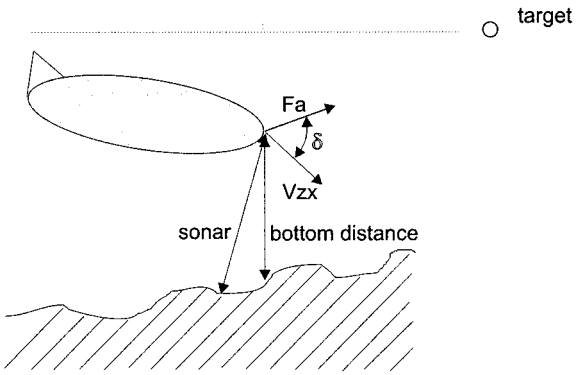

Fig. 14. VFF in the vertical plane.

The state `Trap` activates an escaping procedure based on the WFM. This is a common potential-based path planning technique [4] (see Fig. 13). In detail, the attractive force is computed based on the repulsive force in a way that it *pushes* the vehicle so as to follow the wall.

For the vertical plane, a VFF algorithm based on the sole attractive force $\boldsymbol{F}_a$ is implemented. A sketch of the vehicle under the action of the VFF algorithm in the vertical plane is shown in Fig. 14.

V. Simulation Results

A. Computer Simulations

It is very important to simulate the developed NGS before starting any wet test. Several cases have been analyzed using a simulation tool developed in the MATLAB/SIMULINK environment. The simulations have been run considering the following:

- tracking error of the low-level dynamic control (thruster and the two control surfaces);
- presence of the main hydrodynamic terms, including the current;
- thruster dynamics and saturation, including saturation of the control surfaces;
- noise on sensor readings;
- discretization of the PL with a sampling frequency of 3 Hz;
- frequency acquisition of 12.5 Hz for the vision system, 3 Hz for the vertical sonar and 5 Hz for the panoramic sonar (each incremental movement of the forward sonar is made every 0.2 s).
- specific care to model inaccuracies of the vision system for the pipeline position/orientation detection. In detail, the error is also function of the relative orientation between the pipeline and the cameras thus giving useless information for high angles of incidence.

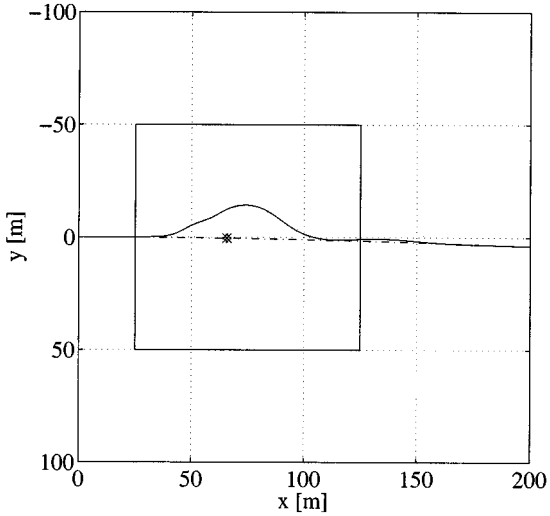

Fig. 15. xy plane of a simulation, Mission 2. The pipeline is represented as a dash-dotted line, the vehicle path as a solid line, and the $\times$s represent the obstacles. The area in the inner square is enlarged in Fig. 16.

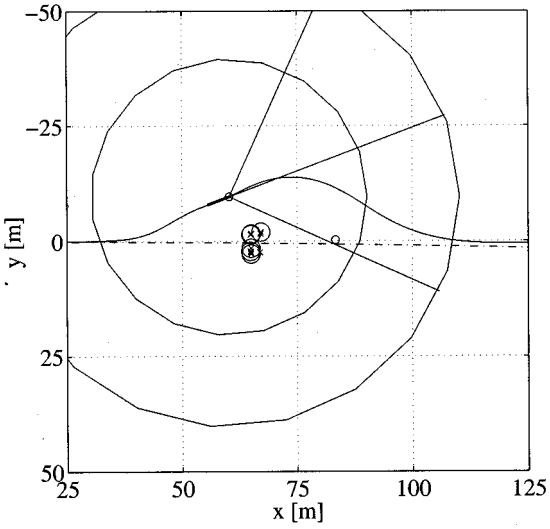

Fig. 16. xy plane of a simulation, Mission 2; detail of Fig. 15. The turn circle and the visibility circle are reported; the circles around the $\times$ symbols (which denote the actual position of the obstacles) represent the obstacles' estimated position; the small circle is the current target. The estimation errors on the obstacle can be recognized; also, the simulated uncertainties cause an error in the target estimation. In this moment, the vehicle has already turned in order to free the overlapping of the two halfcircles.

About 100 simulations, each of them with a pipeline or a desired trajectory of about 4 km, have been run considering a wide range of possible situations. Notice that the scale of the plots does not allow to appreciate the high-frequency simulated noise; moreover, in the vertical plane some discontinuity has been simulated that cannot be observed in the plots for scaling reasons as well. In the following, a sampling of the above simulation results is reported and described to illustrate the performance of the developed NGS in brief.

In Fig. 15, a detail of the xy-plane motion of a simple case of Mission 2 is reported. The solid line is the followed path, the dash-dotted line is the pipeline, and the symbols $\times$ represent obstacles, e.g., the structure in Fig. 2. It can be noted that, during the obstacle avoidance, the vehicle looses visibility of

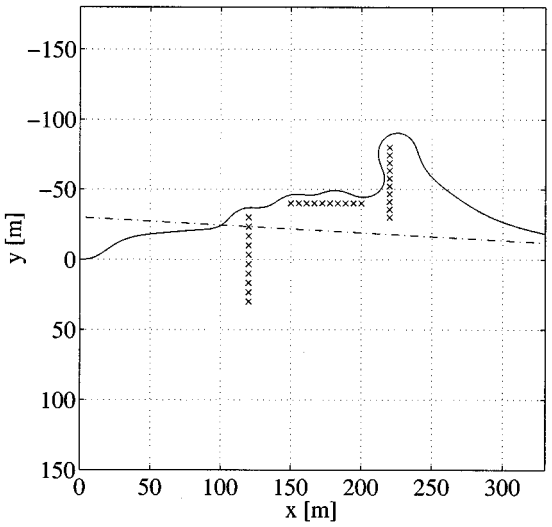

Fig. 17. xy plane of a simulation, Mission 1. The dash-dotted line represents the assigned path.

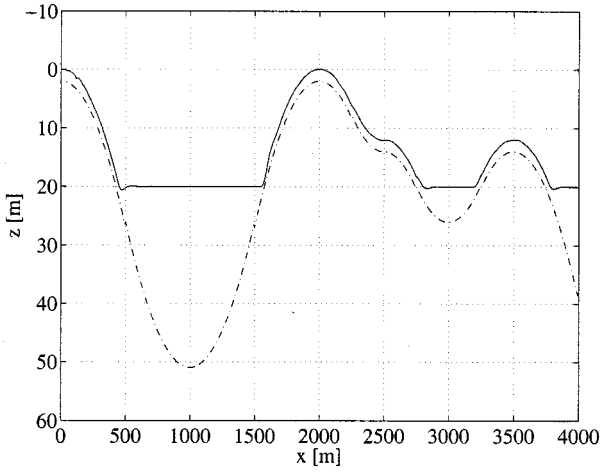

Fig. 18. xz plane of a simulation, Mission 1, constant required depth. The sea bottom is represented as a dash-dotted line, the vehicle path as a solid line. Notice that the scale for x and z is not equal to allow better reading of the vertical component of the motion.

the pipeline and the NGS must reconstruct the pipeline location using the *target* module.

In Fig. 16 some information about the vehicle state during the obstacle avoidance maneuver is reported. The two large circles centered in the vehicle are the turn and the visibility circle; the turn circle has a radius of about 30 m (the vehicle has a minimum turn radius of about 15 m at 1 m/s), and the visibility circle has a radius of 50 m due to the panoramic sonar range. The small circles around the $\times$ symbols are the map information about the obstacles. The three lines with center in the vehicle are the advancing direction and the panoramic sonar range. Finally, the small circle at about 85 m along x is the current target; it can be noted that, due to sensor noise and pipeline curvature, the current target is not exactly on the pipeline.

In Fig. 17 a projection on the xy-plane of the vehicle motion in a case of Mission 1 is reported. In this case, the dash-dotted line represents the assigned path. Note that the starting vehicle position is not onto the assigned path.

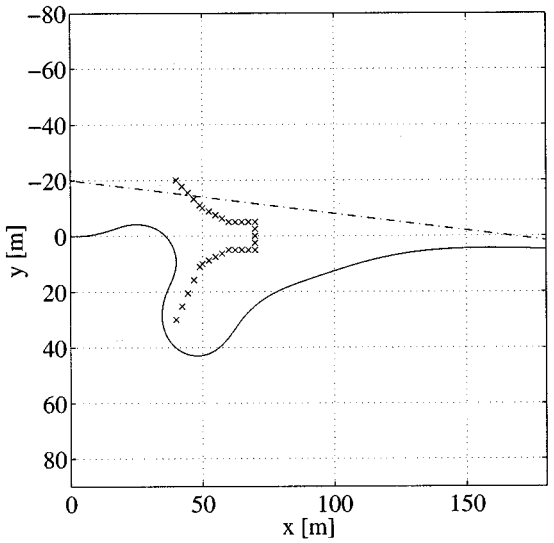

Fig. 19. xy plane of a simulation, canyon-like obstacle. Notice that, if the vehicle enters in the canyon, it does not have enough space the make a 180° escape turn.

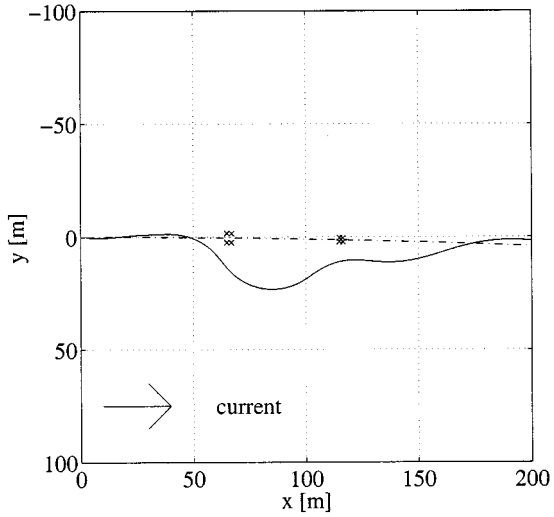

Fig. 20. xy plane of a simulation, mattress and support avoided in presence of a maximum current parallel to the pipeline in the advancing direction. In this case, the vehicle experiences a loss in the maneuverability.

In Fig. 18 the xz-plane motion of a different simulation of Mission 1 is reported. In this case, a constant 20 m depth is required while a safe distance from the bottom is guaranteed. Notice that the scale for x and z is not equal to allow better reading of the vertical component of the motion.

In Fig. 19 the avoidance of a canyon-like obstacle is shown. Due to the panoramic sonar characteristics the vehicle would not be able to foresee the end of the canyon. Moreover, the vehicle cannot pass over the obstacle since the panoramic sonar does not give information about its vertical dimension. The only safe behavior is to avoid entering in such a cluttered space. This is done by the use of the state `Canyon` that, as pointed out in Section IV-B, is used to handle both canyon-like obstacles and obstacles occurring in the forward direction.

Finally, in Fig. 20 a very critical situation is reported. In detail, the vehicle is following the pipeline in presence of the maximum allowed current (0.5 m/s) parallel to the pipeline itself in

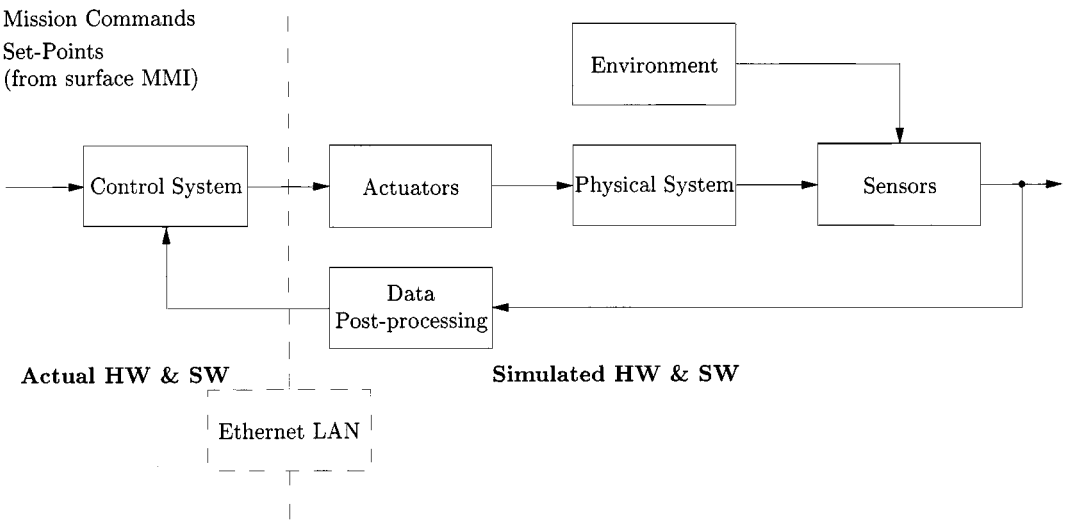

Fig. 21. Hardware-in-the-loop scheme.

the advancing direction. In this case, the maneuverability of the vehicle, which depends on the relative water/vehicle velocity, is drastically reduced. Moreover, during the obstacle avoidance maneuver the minimum turn radius changes with the relative vehicle/current direction and the advancing direction is no longer parallel to the fore-aft direction, leading to the necessity of carefully handling the panoramic sonar. It can also be recognized that the second obstacle is seen by the vehicle only when the approaching phase to the pipeline is commanded. Despite the critical situation, the resulting planned trajectory is very satisfactory.

In the set of all the simulated missions the relative direction between the vehicle and the current has assumed all the possible values. An horizontal current of maximum amplitude normal to the pipeline direction has also been assumed. In this case, the difference between the fore-aft direction and the advancing direction is at its maximum. The control surfaces, the thruster and the sensorial system have guaranteed proper working conditions also in this situation.

B. Hardware-in-the-Loop Simulations

In Fig. 21 a sketch of the testbed used to implement and debug the NGS software is sketched. The *Actuators* block represents the vertical and horizontal rudder actuators and the thruster of the vehicle. The *Physical System* block represents the hydrodynamics of the vehicle. The *Environment* block represents the environment where the vehicle is moving. The *Sensors* block represents the sensors the vehicle is equipped with. The *Data Post-processing* block represents the estimation processing of the vehicle global status (dead reckoning, sensor managing and reading, measure filtering, power supply control, etc.).

The *Control System* block represents the control software and hardware as it will be implemented on the vehicle. In detail, it decodes the mission commands loaded from the surface MMI before vehicle starting (pre-planned mission) or downloaded on-line through an acoustic link. It transforms the high level commands into sequences of piloting commands to be sent one at a time to the NGS developed in this paper. It constantly checks the correct execution of the commands and, in case, it can autonomously manage the following critical situations (the latter feature is under development):

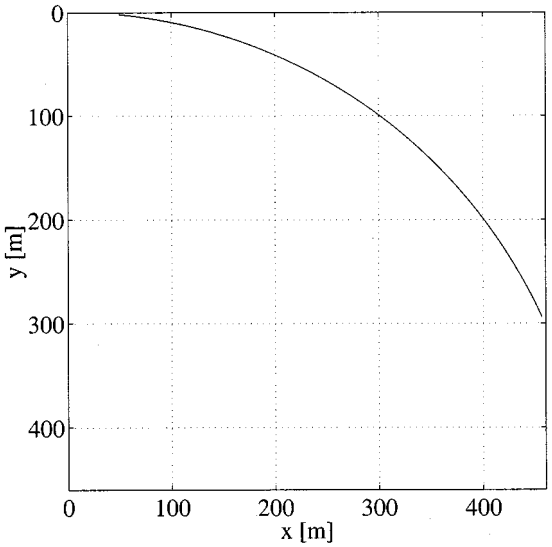

Fig. 22. xy-plane motion of a hardware-in-the-loop simulation, circular pipeline.

- current navigation position much different from the planned one;
- fault on the data post-processing;
- fault on the actuation system.

Several hardware-in-the-loop simulations have been run as a first step toward the implementation of the developed NGS on board.

These simulations are simpler than those reported in the previous Section since they are intended just as a testbed for the software/hardware controller as it will be mounted on RAIS; in fact, validation of the proposed NGS has been extensively pursued via the above numerical simulations. Accordingly, the sensors are considered without noise and no obstacle is present in the environment.

In Fig. 22 the xy view of the motion resulting from a hardware-in-the-loop simulation of Mission 2 is reported; the

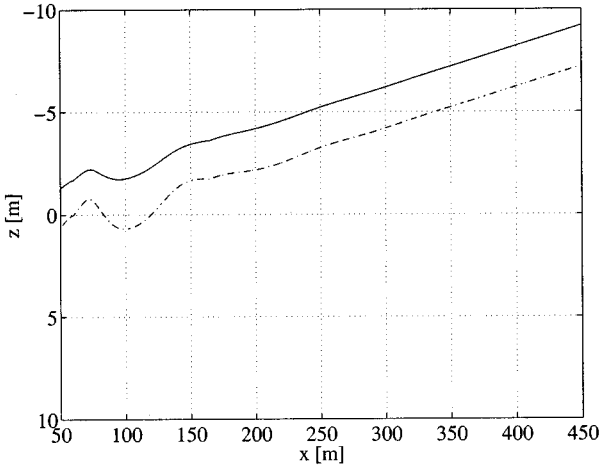

Fig. 23. xz-plane motion of a hardware-in-the-loop simulation, circular pipeline.

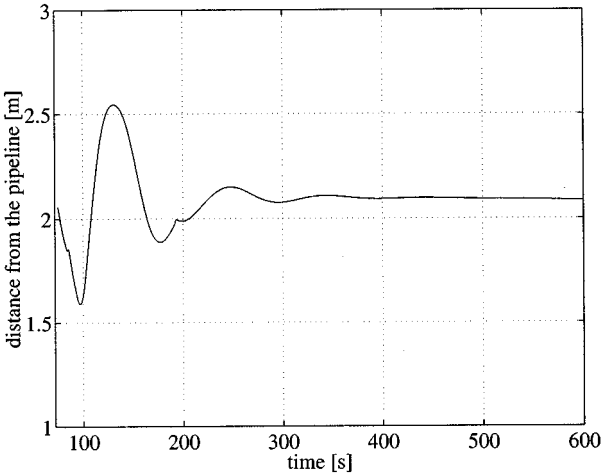

Fig. 24. Time history of the distance from the pipeline in a hardware-in-the-loop simulation, circular pipeline.

pipeline has a circular projection on the horizontal plane. In Fig. 23 the xz view of the motion in the same simulation is shown; it can be seen that the vehicle is following (solid line) the nonregular bottom shape (dash-dotted line). Finally, in Fig. 24 the time history of the distance from the pipeline is reported. The transient is due to two reasons. First, the (unknown) bottom shape is not regular in the beginning. Second, the simulation starts with null initial velocity of the vehicle; in this situation, the rudder and the elevator cannot provide the necessary control action which are instead possible later when the vehicle velocity reaches the cruise value.

VI. Conclusions

An NGS for real-time path planning and obstacle avoidance of an autonomous underwater vehicle has been presented in this paper. This is based on a PL that generates the motion reference trajectory needed to accomplish two missions: pre-deployment survey of sea bottom and visual inspection of sealines. The PL has been designed taking into account the drastic constraints of the underwater environment: uncertainty in the sea bottom profile, presence of unknown obstacles, presence of the ocean current, uncertainty in the sealine position. Extensive numerical and hardware-in-the-loop simulations, taking into account vehicle and thruster dynamics and sensor noise, have proven the effectiveness of the proposed NGS.

References

[1] F. Affaitati, "ENI-MISM. Sistema di Rivelazione Ostacoli, Prove Funzionali," (in Italian), , Venezia, Tecnomare Internal Rep. B0704-REL-A210-006.0, pt. I, 1998.

[2] G. Antonelli and S. Chiaverini, "Real-time motion planning for autonomous underwater vehicles," in *Prep. 14th IFAC World Congr.*, vol. B, Beijing, PRC, July 1999, pp. 59–64.

[3] G. Antonelli, S. Chiaverini, R. Finotello, and E. Morgavi, "Real-time path planning and obstacle avoidance for an autonomous underwater vehicle," in *Proc. 1999 IEEE Int. Conf. Robotics and Automation*, Detroit, MI, May 1999, pp. 78–83.

[4] J. Borenstein and Y. Koren, "Real-time obstacle avoidance for fast mobile robots," *IEEE Trans Syst., Man, Cybern.*, vol. 19, pp. 1179–1187, 1989.

[5] J. Borenstein and Y. Koren, "Real-time obstacle avoidance for fast mobile robots in cluttered environments," in *Proc. IEEE Int.Conf. Robotics and Automation*, Cincinnati, OH, May 1990, pp. 572–577.

[6] J. Borenstein and Y. Koren, "The vector field histogram-fast obstacle avoidance for mobile robots," *IEEE Trans. Robot. Automat.*, vol. 7, pp. 278–288, 1991.

[7] R. A. Brooks, "A robust layered control system for a mobile robot," *IEEE Trans. Robot. Automat.*, vol. 2, pp. 14–23, 1986.

[8] G. Conte, S. Zanoli, E. Pascucci, and A. Radicioni, "A navigation and inspection system for underwater survey vehicles," in *Prep. 4th IFAC Symp. Robot Control*, Capri, Italy, Sept. 1994, pp. 1025–1030.

[9] H. J. S. Feder, J. J. Leonard, and C. M. Smith, "Incorporating environmental measurements in navigation," in *Proc. 1998 Workshop Autonomous Underwater Vehicles*, July 1998, pp. 115–122.

[10] G. L. Foresti, S. Gentili, and M. Zampato, "A vision based system for autonomous underwater vehicle navigation," in *Proc. IEEE Oceans '98*, Nice, Italy, Sept. 1998, pp. 195–199.

[11] T. Fossen, *Guidance and Control of Ocean Vehicles*. Chichester, UK: Wiley, 1994.

[12] S. Hert, S. Tiwari, and V. Lumelsky, *A Terrain-Covering Algorithm for an AUV Underwater Robots*, Yuh, Ura, and Bekey, Eds. Boston, MA: Kluwer , 1996, pp. 17–45.

[13] J. C. Hyland and F. J. Taylor, "Mine avoidance techniques for underwater vehicles," *IEEE J. Oceanic Eng.*, vol. 18, pp. 340–350, 1993.

[14] Y. Koren and J. Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," in *Proc. IEEE Int. Conf. Robotics and Automation*, Sacramento, CA, Apr. 1991, pp. 1398–1404.

[15] J. C. Latombe, *Robot Motion Planning*. Boston, MA: Kluwer Academic, 1991.

[16] V. Lumelsky and T. Skewis, "Incorporating range sensing in the robot navigation function," *IEEE Trans. Syst., Man, Cybern.*, vol. 20, pp. 1058–1069, 1990.

[17] D. B. Marco, A. J. Healey, and R. B. McGhee, "Autonomous underwater vehicles: Hybrid control of mission and motion," in *Underwater Robots*, Yuh, Ura, and Bekey, Eds. Boston, MA: Kluwer, 1996, pp. 95–112.

[18] V. Rigaud *et al.*, "UNION: UNderwater Intelligent Operation and Navigation," *IEEE Robot. Automat. Mag.*, pp. 25–35, Mar. 1998.

[19] C. M. Smith, J. J. Leonard, and H. J. S. Feder, "Making difficult decisions autonomously: The impact of integrated mapping and navigation," in *Proc. Workshop Autonomous Underwater Vehicles*, July 1998, pp. 123–132.

[20] I. Ulrich and J. Borenstein, "VFH+: Reliable obstacle avoidance for fast mobile robots," in *Proc. IEEE Int. Conf. Robotics and Automation*, Leuven, Belgium, May 1998, pp. 1572–1577.

[21] K. P. Valavanis, D. Gracanin, M. Matijasevic, R. Kolluru, and G. A. Demetriou, "Control architecture for autonomous underwater vehicles," *IEEE Control Syst. Mag.*, pp. 48–64, Dec. 1997.

[22] C. Vasudevan and K. Ganesan, "Case-based path planning for autonomous underwater vehicles," in *Underwater Robots*, Yuh, Ura, and Bekey, Eds. Boston, MA: Kluwer, 1996, pp. 1–15.

[23] C. W. Warren, "A technique for autonomous underwater vehicle route planning," *IEEE J. Oceanic Eng.*, vol. 15, pp. 199–204, 1990.

[24] D. R. Yoerger, A. M. Bradley, B. B. Walden, H. Singh, and R. Bachmayer, "Surveying a subsea lava flow using the autonomous bentic explorer (ABE)," in *Pre-Proc. 6th Int. Advanced Robotics Program*, 1996, pp. 1–21.

Gianluca Antonelli was born in Rome, Italy, in 1970. He received the "Laurea" in electronic engineering and "Research Doctorate" degree in electronic engineering and computer science from the University of Naples, Naples, Italy, in 1995, and 2000, respectively.

He currently is a Post-Doctoral Fellow at the University of Cassino, Cassino, Italy, where he is investigating simulation and control of underwater robotic systems. His research interests include force/motion control of robot manipulators, nonlinear control, path planning and obstacle avoidance for autonomous vehicles, identification. From April to July 1996 he was a Visiting Scholar with the Université Catholique de Louvain, CESAME, Louvain-la-Neuve, Belgium, from September to December 1996 with the Katholieke Universiteit Leuven, Division PMA, Heverlee (Leuven), Belgium, and from November 1998 to April 1999 with the University of Hawaii at Manoa, Autonomous Systems Laboratory, Honolulu, HI.

Stefano Chiaverini was born in Naples, Italy, in 1961. He received the "Laurea" degree and the "Research Doctorate" degree in electronic engineering from the University of Naples, Naples, Italy, in 1986 and 1990, respectively.

From 1990 to 1998 he has been at the Department of Computer and Systems Engineering of the University of Naples. He is currently an Associate Professor of Automatic Control in the Faculty of Engineering of the University of Cassino. His research interests include manipulator inverse kinematics techniques, redundant manipulator control, force/motion control of manipulators, cooperative robot manipulation, and underwater robotics.

He has published about 100 journal and conference papers and he is co-editor of the book "Complex Robotic Systems" (London: Springer-Verlag, 1998). He is currently serving as an Associate Editor of the IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION.

Roberto Finotello was born in Venice, Italy, in 1963. He received the "Laurea" degree in electronic engineering from the University of Padoa, Padoa, Italy, in 1987.

In 1989 he joined Tecnomare S.p.A., Venice, Italy as a System Engineer in where he is since 1998, a Project Engineer in the Robotics and Space Department. He was involved in development of autonomous underwater vehicles. He is currently working in the development of an autonomous robotic system for space environment.

Riccardo Schiavon was born in Padoa, Italy, in 1967. He received the "Laurea" degree in electronic engineering at Padoa University, Padoa, Italy, in 1994.

He is a Lead Engineer at the Robotics and Subsea Department of Tecnomare S.p.A., Venice, Italy, a company for the development of the marine technologies. He has worked in the field of control system, architectures and real time software development for underwater robotic systems. He is currently involved in the final development of an autonomous underwater vehicle for Antarctica research.