User-assisted trajectory generation of underwater vehicles

Sérgio Loureiro Fraga, João Borges de Sousa, Fernando Lobo Pereira
Instituto de Sistemas e Robótica - Porto
Faculdade de Engenharia da Universidade do Porto
Rua Dr. Roberto Frias
4200-465 Porto, Portugal
E-mail: {slfraga,jtasso,flp}@fe.up.pt

***Abstract*— This paper presents the user-assisted trajectory generation module of a control system for autonomous and remotely operated vehicles. The control system is composed of three main modules: user interface, trajectory generation and feedback control. The paper reports the integrated design of the user interface and of the trajectory generation module and discusses how this design can impact the overall system performance. The approach to trajectory generation is based on the techniques from differential flatness.**

I. Introduction

Underwater vehicles are designed to perform missions, which are composed of several tasks. These tasks may involve the inspection of underwater structures, data collection, map building, etc. The problem of designing trajectories for each mission requires the intervention of a mission expert, to capture the domain knowledge, and the consideration of dynamic models for each vehicle. The quality of the models is strongly related to the performance of the trajectory, and hence task performance.

Traditionally, the mission expert is not an expert in modeling and control. Hence, when it comes to trajectory generation the mission expert does not take into account the dynamic model of a vehicle and it is up to the control system to handle this additional "difficulty". This obviously results in poor performance and, even worse, in undesirable behavior when it comes to mission execution i.e., the trajectory specification was not correct.

This paper presents a user-assisted trajectory generation system for autonomous and remotely operated vehicles. We provide solutions for two kinds of underwater vehicles: autonomous underwater vehicles (AUV) and remotely operated vehicles (ROV). In what concerns the AUV, underwater communications pose serious constraints to real-time control. Hence, the mission should be specified such that the vehicle can decide when to emerge in order to verify if some mission update is required or, eventually, to send relevant information to the mission expert, during the execution of the mission. In the ROV, the communication between the pilot and the vehicle is not a problem since there is a point-to-point link between them. In this case, the pilot is able to directly control the vehicle through a joystick or to issue high-level commands for the execution of complex maneuvers.

Our approach divides the project of the control system in three main modules: user interface, trajectory generation and feedback control. In this paper we report the integrated design of the user interface and the trajectory generation module, focusing on the user-assisted features of the design. We show how this design can impact the overall system performance.

The approach is based on the techniques from differential flatness. In a differential flat system there is an algebraic relation between the trajectory specification in the output space and the state and control inputs. This means that these are uniquely defined if the output trajectory is given. The problem of motion planning of a rigid body with kinematics constraints is then reduced to the problem of motion planning without differential constraints.

This paper is organized as follows. Section II presents the applications that motivated the present work. Section III discusses the trajectory generation project by presenting the model of the vehicles and by introducing differentially flat systems. Section IV presents the project of a user-interface to the trajectory generation module. Finally, section V presents some concluding remarks.

II. Applications

The main motivation for the development of a user-assisted trajectory generation system comes from two classes of underwater vehicles: Autonomous Underwater Vehicles (AUV) and Remotely Operated Vehicles (ROV). The purpose of this section is to provide an overview of these vehicles and outline the types of missions they are able to perform. We will focus on the vehicles of the Underwater Systems and Technology Laboratory (USTL) from Porto University.

Isurus (figure 1) is a REMUS (Remote Environment Measuring UnitS) class AUV, built by the Woods Hole Oceanographic Institution, MA, USA. These vehicles are low cost, lightweight, and specially designed for coastal waters monitoring . The reduced weight and dimensions makes them extremely easy to handle, requiring no special equipment for launching and recovery.

Isurus has a torpedo shaped hull, about 1.6 meters long, with a diameter of 20 cm and weighting about 35 kg in air. The maximum forward speed is 4 knots, being the best energy efficiency achieved at about 2 knots. At this velocity, the

0-933957-31-9

Fig. 1. ISURUS autonomous underwater vehicle

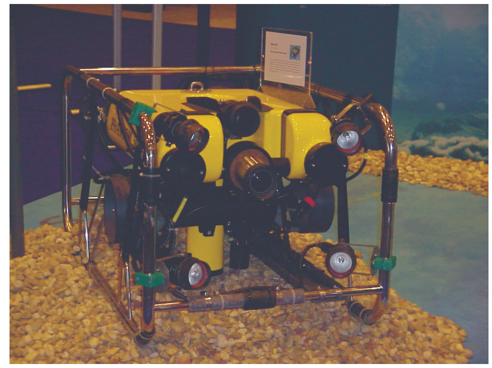

Fig. 2. The IES Remotely Operated Vehicle

energy provided by a set of rechargeable Lithium-Ion batteries may last for over 20 hours (i.e., over 40 nautical miles). Isurus can be configured for different types of missions and, currently, integrates all of the technologies developed at USTL [1]. The basic configuration is summarized in the next table:

Sensors	OS200 CTD (Ocean sensors) Optical Backscatter (Wet Labs) SideScan Sonar (Marine Sonics) Altimeter (Imagenex)
Navigation	TCM2 Digital Compass (PNI) LBL Acoustic Beacons 20-30 Khz
Actuation	Propeller Horizontal and vertical fins
Computer system	PC104 Technology QNX real-time operating system

The first operational missions with Isurus took place in 1998, in the estuary of the river Minho, in the northern border between Portugal and Spain. The typical mission lasted over one hour and consisted of Conductivity, Temperature, Depth (CTD), and bathymetric data collection. These missions demonstrated the reliability and the operational effectiveness of this vehicle. Since then several other missions have been performed on different scenarios. Last summer, Isurus mapped the plume generated by a sea sewage outfall, 3 km off the Portuguese coast. The vehicle successfully performed a 2 hours mission, under very severe sea conditions. This innovative mission profile is now part of the monitoring plan for this sea outfall.

Remotely Operated Vehicles are unmanned and tethered submersibles piloted by a human operator. A ROV system usually consists of the ROV, a tether and the tether management system, and an operator console. The tether is used for information and power transmission. Typically ROVs are used in the real-time visual inspection of underwater structures, such as ship hulls, port structures, underwater pipe and cable systems.

The USTL has been developing and operating an ROV (see figure 2) for the last 3 years. This vehicle is being developed for the project Inspection of Underwater Structures (IES) [2][1]. The IES project concerns the design and implementation of an advanced low cost system for the inspection of underwater structures based on an ROV. During the last phase of the project the IES system has been extensively used for several different inspection missions in the Leixões port. The IES system has successfully operated in near-zero visibility, taking high quality video footage of several underwater structures. In 2003 the system will be used in commercial operations ranging from inspection in harbors and dams to coastal underwater archaeological missions.

The characteristics of the IES system are summarized in the following table:

Inspection system	Video Camera, Inspector (ROS) Pan and Tilt unit (Imenco) 600W of light (DSP&L)
Navigation	Argonaut Doppler Velocimeter (Sontek) Inertial Unit, HG1700 (Honeywell) Digital Compass, TCM2 (PNI) Acoustic Beacons, LBL 20-30KHz
Thrusters	4 DC Motors, 120V, 1/8 HP (Bodine)
Umbilical	Neutrally buoyant in fresh water Video, Ethernet and power (Falmat) 300 meters long Electrical slip ring at the surface
Computer System	PC-104 Technology QNX Real Time Operating System CAN Local Bus Ethernet interface with the console

The main innovations of the IES project with respect to commercially available ROV solutions are:

- On-board power and computer systems.
- Two modes of operation: tele-operation and tele-programming. The tele-operation mode is a standard feature in ROV systems. The tele-programming mode enables the operator to program automated operations, such as trajectory or path tracking.
- Integrated navigation.
- PC-based control.

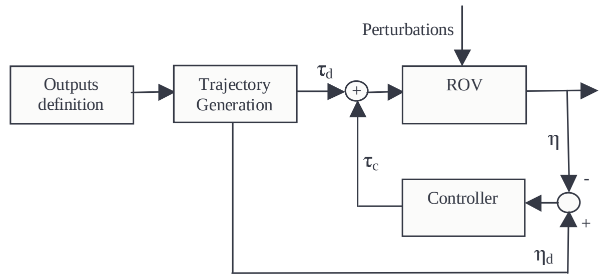

Fig. 3. Modules of the trajectory generation system

III. Trajectory generation design

A. Control system

We propose a three-layered architecture for the control system (figure 3):

- User interface for output specification. This module has a graphical user interface to assist the operator in the specification of the desired values for the outputs. Upon validation, the pilot's objectives (represented as system outputs) are sent to the trajectory generation module.
- Trajectory generation. This module computes a feasible trajectory. The output of this module is a complete description of the desired state and input of the system.
- Controller. This module controls the actuators to track the trajectory defined by the trajectory generation module even in the presence of external disturbances and modeling errors.

In this paper we describe in more detail the trajectory generation and user interface modules.

B. Models

The motions of the AUVs and of the ROVs are described using either an inertial coordinate system XYZ or a body fixed coordinate system $X_0Y_0Z_0$. For our purposes, we consider an earth-fixed coordinated system as inertial and the body-fixed coordinate system has its origin in the vehicle's center of mass and the axis directions coincide with the vehicle's principal inertial axis:

- X_o - longitudinal axis (from aft to fore)
- Y_o - transverse axis (from port side to starboard)
- Z_o - vertical axis (from top to bottom).

Position and velocity are represented in the inertial and body fixed coordinate system, respectively. We adopted the notation from "The Society of Naval Architects and Marine Engineers (SNAME)" (1950). According to SNAME, the motion of underwater vehicles in six degrees of freedom can be described by the following vectors:

$$\begin{array}{lll} \eta = \left[\eta_1^T, \eta_2^T\right]^T; & \eta_1 = [x, y, z]^T; & \eta_2 = [\phi, \theta, \psi]^T; \\ \nu = \left[\nu_1^T, \nu_2^T\right]^T; & \nu_1 = [u, v, w]^T; & \nu_2 = [p, q, r]^T; \\ \tau = \left[\tau_1^T, \tau_2^T\right]^T; & \tau_1 = [X, Y, Z]^T; & \tau_2 = [K, M, N]^T. \end{array}$$

where η represents the inertial coordinates of the vehicle, ν represents the linear and angular velocities in body fixed coordinates and, finally, τ are the forces and moments applied on the vehicle in body fixed coordinates.

The kinematical equations relating the state variables in the inertial and body fixed coordinate system are given by:

$$\begin{bmatrix} \dot{\eta}_1 \\ \dot{\eta}_2 \end{bmatrix} = \begin{bmatrix} J_1(\eta_2) & 0_{3\times 3} \\ 0_{3\times 3} & J_2(\eta_2) \end{bmatrix} \begin{bmatrix} \nu_1 \\ \nu_2 \end{bmatrix} \Leftrightarrow \dot{\eta} = J(\eta)\nu \quad (1)$$

where J_1 and J_2 are transformation matrix (details in [3]). The matrix J_2 has a singularity for $\theta = \pm\frac{\pi}{2}$ and is not a rotational matrix, i.e. it does not belong to $SO(3)$. This singularity is not a problem for vehicles' control since the vehicles never reach a state where $\theta = \pm\frac{\pi}{2}$. J_1 matrix is a rotational matrix, i.e. $J_1 \in SO(3)$.

The external forces acting on the vehicle are classified in three categories [3][4]: radiation-induced forces τ_H (added inertia, hydrodynamic damping, restoring forces), environmental forces τ_E (underwater currents, waves, wind) and propulsion forces τ (thrusters and propeller forces, control surfaces and rudder forces).

The hydrodynamic forces τ_H are given by:

$$\tau_H = -M_A\dot{\nu} - C_A(\nu)\nu - D(\nu)v - g(\eta). \quad (2)$$

The term $-M_A\dot{\nu} - C_A(\nu)\nu$ models the added mass due to the inertia of the surrounding fluid, while the term $-D(\nu)\nu$ represents the total hydrodynamic damping [3]. Finally, the component $-g(\eta)$ represents the restoring forces due to the vehicles' weight and buoyancy as follows:

$$g(\eta) = \mathrm{col}(0, 0, 0, -z_B B\cos\theta\sin\phi, -z_B B\sin\theta, 0), \quad (3)$$

where z_B represents the distance between the application point of the weight and the buoyancy force and B is the impulsion force, which is equal to the weight W for these vehicles.

The resulting force acting on the vehicle is given by:

$$\tau_{RB} = \tau_H + \tau_E + \tau \quad (4)$$

We are now able to write the 6 dof dynamic model in body-fixed coordinates:

$$M\dot{\nu} + C(\nu)\nu + D(\nu)\nu + g(\eta) = \tau_E + \tau \quad (5)$$

where $M = M_A + M_{RB}$ and $C(\nu) = C_{RB}(\nu) + C_A(\nu)$. A full description of these matrices can be found in [3].

The kinematics equation 1 is used to write this model in earth-fixed coordinates:

$$M_\eta(\eta)\ddot{\eta} + C_\eta(\nu,\eta)\dot{\eta} + D_\eta(\nu,\eta)\dot{\eta} + g_\eta(\eta) = \tau_\eta. \quad (6)$$

where

$$\begin{array}{rcl} M_\eta(\eta) & = & J^{-T}(\eta)MJ^{-1}(\eta) \\ C_\eta(\nu,\eta) & = & J^{-T}(\eta)\left[C(\nu) - MJ^{-1}(\eta)\dot{J}(\eta)\right]J^{-1}(\eta) \\ D_\eta(\nu,\eta) & = & J^{-T}(\eta)D(\nu)J^{-1}(\eta) \\ g_\eta(\eta) & = & J^{-T}(\eta)g(\eta) \\ \tau_\eta(\eta) & = & J^{-T}(\eta)\tau. \end{array} \quad (7)$$

These equations do not include models of actuators and of environmental forces such as underwater currents, waves and wind. Our results can be easily extended to encompass these models.

Both the ROV and the AUV operated by USTL are under-actuated.

The AUV has one propeller placed on the back with two rudders associated with it. The rudders control pitch and yaw. The force system τ generated by the propeller and the rudders is:

$$\begin{bmatrix} X \\ Y \\ Z \\ K \\ M \\ N \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & -d \\ 0 & d & 0 \end{bmatrix} . \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix} . \quad (8)$$

where F_1 is the force applied by the propeller, F_2 and F_3 are forces applied by the rudders and d is the distance between the vehicle's center of mass and the rudders.

The ROV has four thrusters. The relationship between the forces applied by the thrusters (F_i) and τ is given by (values in SI):

$$\begin{bmatrix} X \\ Y \\ Z \\ K \\ M \\ N \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -0,07 & -0,02 \\ -0,10 & -0,10 & 0 & 0,02 \\ 0,18 & -0,22 & 0,14 & 0 \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} \quad (9)$$

Both the AUV and the ROV are under-actuated (see equations 8 and 9). However, and except for roll in the AUV, there are couplings between the force system imposed by the actuators and all of the degrees of freedom. This relates to the problem of global controllability. Informally, a system, in particular an under-actuated system, is globally controllable if it can reach any configuration in the state-space. In practice, this is the reason why trajectory generation for globally controllable under-actuated systems is a challenging task: we have to compute a suitable input τ and a state v trajectory to implement a desired output trajectory y_d. This is not easy even if we have a desired trajectory for the state because we have less inputs than state variables. We cannot solve this problem by simple solving of a nonlinear system of equations. Moreover, not every output trajectory is feasible i.e., we cannot find a input to implement it. The system is controllable only under suitable motion planning. Systems that are controllable and under-actuated are known as systems with nonholonomic constraints [5][6]. This is exactly why we need to assist the pilot or the operator of systems such as our vehicles in the process of trajectory generation. This will be addressed in the next section.

The problem of global controllability for non-linear systems is studied using results from nonlinear analysis, namely the Chow and Frobenious theorems [7][8]. Basically, these theorems study the involution of distributions spanned by the vector fields associated with each actuator and the vector field of the system dynamics. In order to obtain those vector fields we write the vehicle model in the following form [8]:

$$\begin{aligned} \dot{\nu} &= f(\nu) + \sum_{i=1}^{m} g_i(\nu).\tau_i \\ y_j &= h_j(\nu) \quad 1 \le j \le l \end{aligned} \quad (10)$$

where $f, g_1, ..., g_m$ are smooth vector fields defined on R^n and h_i are smooth real-valued functions defined on R^n. The first relation represents a differential equation on R^n, and $\dot{\nu}$ is the tangent vector at the point ν to the smooth curve which characterizes the solution for some initial condition. y_j represents the system output.

C. Differential flatness

In order to assist an operator in the process of trajectory generation we need to be able to: 1) compute the input and the state of the system from an output trajectory and; 2) specify feasible trajectories. The solution for these two problems relies on the properties of differentially flat systems. A system is differentially flat [9] if it is possible to find a set of variables (equal in number of the inputs), called flat outputs, such that all state variables and input become defined from them without integration. Let $\eta \in R^n$ be the system state and $\tau \in R^m$ its input, then the system is flat if it is possible to find flat outputs $z \in R^m$ in the form:

$$z = \mathbf{Z}(\eta, \tau, \dot{\tau}, ..., \tau^{(l)}) \quad (11)$$

such that

$$\begin{aligned} \eta &= \boldsymbol{\eta}(z, \dot{z}, ..., z^{(q)}) \\ \tau &= \boldsymbol{\tau}(z, \dot{z}, ..., z^{(q)}) \end{aligned} \quad (12)$$

where $\mathbf{Z}, \boldsymbol{\eta}, \boldsymbol{\tau}$ are 1-1 smooth mapping functions.

The trajectory of the flat outputs has no differential constraints – we can specify the trajectory freely in the state space. Thus, we compute the input and the state by solving a nonlinear system of equations.

Differentially flat systems are of practical interest when the flat outputs can be used to express the desired behavior of the system, i.e. there is a physical meaning attached to them. Finding flat outputs with those features is one of the challenges of this approach since there is not any systematic method to do this. Reference [10] contains a catalog of differentially flat systems.

As an example consider an ellipsoid shaped rigid body (similar to the AUV presented before) with the actuators mounted as in our AUV. This is differentially flat system [11] where the flat outputs correspond to three dimensional point located inside the rigid body (in body-fixed coordinates $z = (-\frac{I_y + I_z}{m.d}, 0, 0)$).

The ROV is also a differentially flat system [12][13] since for this vehicle roll and pitch are stable. This means that we can treat this vehicle as a fully actuated system and the flat outputs coincide with x, y, z, ψ. The advantages arising from the property of differential flatness are more relevant for the AUV where nonholonomic constraints are present.

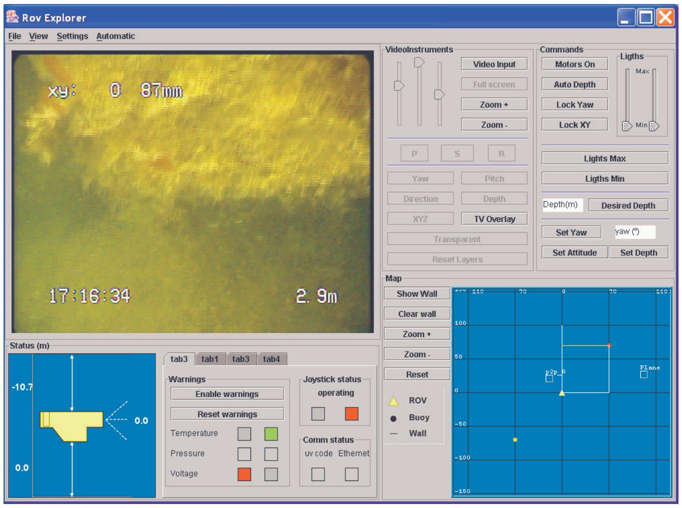

Fig. 4. Graphical user interface for tele-operation mode

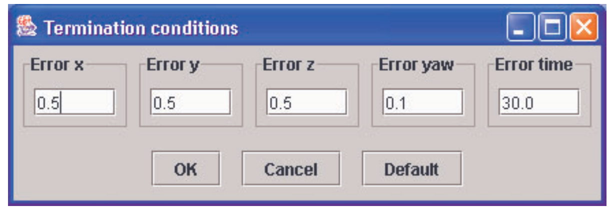

Fig. 5. Termination conditions for point to point maneuver

In terms of trajectory specification the only constraints imposed on the flat outputs refer to the initial and final configurations and smoothness. Other constraints, such as input saturation, can be converted to the flat outputs trajectories by imposing bounds on their curvatures and derivatives.

IV. User-assisted Trajectory Generation

It is now clear why we need to assist the pilot in the process of trajectory generation. In practice, the process of trajectory generation starts when the pilot wants to specify motions for a specific vehicle. The motions may be specified as waypoints, joystick commands or even more abstract objects such as pre-programmed maneuvers. We propose three criteria for capturing the pilot's objectives efficiently:

- Abstraction: this criteria permits introduction of pilot's objectives in a complex structure in the way he perceives it.
- Contextual environment: this is an extension of the previous criteria. The pilot should express his objectives with no concern for external factors such as initial conditions or weather conditions. These factors should only be considered during the trajectory generation process.
- Granularity: this criterion selects from the objectives introduced by the pilot the ones that are really necessary for trajectory generation. This procedure reduces the complexity of the trajectory generation.

After the introduction of the objectives, the automated platform is now able to compute the trajectories for the vehicles. During trajectory generation it may happen that not all pilot's objectives can be satisfied. Thus, the automated platform must interact with the user to achieve a trade-off between the initial desired objectives and the possible ones. This may happen for instance when the external conditions, such as aquatic currents, do not allow to achieve the desired configuration. In this situation the user-interface must negotiate other alternatives with the pilot.

The three criteria presented were implemented in the user interface of the ROV. Our propose now is to present an implementation of what was said. There are two modes of operation for the ROV: tele-operation and tele-programming. In tele-operation the pilot sends velocity commands through a joystick. The trajectory generation module transforms those commands into suitable inputs for each thruster. This way we conform to the abstraction and contextual environment criteria since the pilot sends velocity commands without considering currents, nonholonomic constraints or asymmetries in the way the thrusters are mounted. The user interface for tele-operation presents the following information to the pilot :

- Images – from the on-board camera.
- State of the vehicle in both inertial and body-fixed referential – inertial data is useful for navigation while body-fixed data gives a measurement of the velocity of the vehicle.
- Positioning maps – to display the location of the vehicle. External features such as ships, walls or docks are also plotted in this map. This gives a synthetic image of the operational scenario.

In addition to the joystick commands the pilot is also able to operate in semi-operated modes. For example, he may send velocity commands while the control system regulates the depth of the vehicle or maintains its yaw. These features contribute for a higher degree of abstraction during the operation of the ROV thus improving the operational efficiency. In figure 4 we observe the user-interface for this operational mode.

In tele-programming mode the operation is slightly different. In this mode the role of the user-interface is to support the construct a motion plan for the ROV. A motion plan is an abstract representation of the vehicle behavior and takes the form of motion and behavior patterns which we will denominate by maneuvers. This canonical and discrete form to describe the vehicle's behavior is the natural way for humans to interact with automated platforms. The set of available maneuvers defines the vehicle's capabilities. The ROV maneuvers are: move between two different configurations (point to point), follow a trajectory or a path, and restrict the ROV motion to a plane [2]. In this operational mode the only thing the pilot has to do is to define the parameters of each maneuver and the sequence of maneuvers. This is done with a graphical interface where it is shown the parameters available to fill and the range they can take, a view of the scheduled maneuvers and a map where are plotted the trajectories of each maneuver.

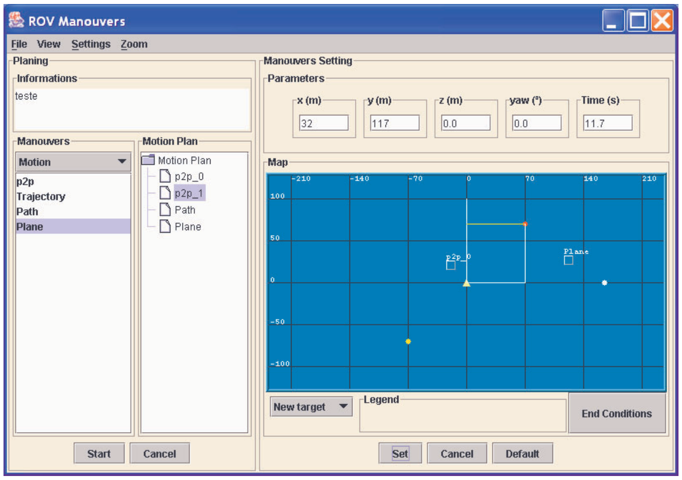

Fig. 6. Graphical user-interface for the tele-programming mode

For instance, the parameters available for the point to point maneuver are the inertial coordinates x, y, z, ψ at the instant t (also a parameter). The pilot also specifies the termination conditions for a maneuver. For example, in the point to point maneuver the termination conditions are the maximal error between the desired values and the sensed ones (figure 5). When there is more than one maneuver the user-interface is able to schedule the maneuvers by checking their termination and initializes the next maneuver of the motion plan. This is done till the last maneuver be completed. When a maneuver cannot finish successfully the user-interface warns the pilot and terminates the motion plan.

If the pilot specifies maneuver parameters that cannot be implemented by any feasible trajectory, the user-interface suggest others parameters indicating the range that the variables can take. In figure 6 we can see the user-interface for the tele-programming mode.

Trajectory generation is based on the properties of differentially flat systems, as is the case of the ROV and of the AUV. The parameters for each maneuver are transformed onto desired outputs. When these outputs are also flat outputs it is quite easy to generate a trajectory for the state and for the inputs. Although the operator has not knowledge about nonlinear systems he is actually defining a trajectory for the flat outputs. This, by itself, allows a high degree of abstraction for the pilot, which is one of the requirements for this kind of system.

V. Conclusions

This paper reports an user-assisted framework for trajectory generation for two classes of underwater vehicles. The framework takes advantage of the differential flatness of both vehicles. The framework assists a pilot with no prior knowledge of nonlinear systems in the process of designing feasible trajectories.

Acknowledgments

This material is based upon work funded by the Programa PRAXIS XXI - Medida 3.1b) (IES project) and by Ministério da Defesa, Portugal.

References

[1] Fernando Lobo Pereira, "Control design for autonomous vehicles: A dynamic optimization perspective," in *Fundamental Issues in Control*. 2001, vol. 7, European Journal of Control.

[2] Sérgio Loureiro Fraga, João Sousa, Anouck Girard, and Alfredo Martins, "An automated maneuver control framework for a remotely operated vehicle," in *MTS/IEEE Oceans 2001*. 2001, IEEE.

[3] Thor I. Fossen, *Guidance and Control of Ocean Vehicles*, John Wiley and Sons, 1995.

[4] Rui Manuel Ferreira Gomes, "Modeling and control of underwater vehicles," M.S. thesis, Faculdade de Engenharia da Universidade do Porto, 2002.

[5] Jean-Paul Laumond, *Robot Motion Planning and Control*, Springer, 1998.

[6] Jean-Claude Latombe, *Robot Motion Planning*, KAP, 1993.

[7] Richard Murray, Zexiang Li, and S. Sastry, *Mathematical Introduction to Robotic Manipulation*, CRC Press, 1994.

[8] Alberto Isidori, *Nonlinear Control Systems*, Springer, 1989.

[9] Michiel Van Nieuwstadt and Richard Murray, "Approximate trajectory generation for differentially flat systems with zero dynamics," in *IEEE Conference on Decision and Control, New Orleans*, 1995.

[10] Richard Murray, Muruhan Rathinam, and Willem Sluis, "Differential flatness of mechanical control systems: A catalog of prototype systems," in *ASME International Mechanical Engineering Congress and Exposition*, 1995.

[11] Michiel J. Van Nieuwstadt, *Trajectory Generation for Nonlinear Control Systems*, Ph.D. thesis, California Institute of Technology, Pasadena, California, 1997.

[12] Sérgio Loureiro Fraga, João Borges Sousa, and Fernando Lobo Pereira, "Trajectory generation for a remotely operated vehicle," in *IEE European Control Conference*, September 2003.

[13] Sérgio Loureiro Fraga, "Trajectory generation - application to underwater vehicles," M.S. thesis, Faculdade de Engenharia da Universidade do Porto, 2002.