A Behavior-Based Mission Planner for Cooperative Autonomous Underwater Vehicles

Laura Sorbi⁽¹⁾, Graziano Pio De Capua^{(1)†}, Jean-Guy Fontaine⁽²⁾ and Laura Toni⁽¹⁾

⁽¹⁾ Italian Institute of Technology (IIT), Genova, Italy.

⁽²⁾ Centre de Robotique Intégrée d'Ile de France C.R.I.I.F

E-mail: {laura.sorbi, grazianopio.decapua}@iit.it, fontaine@criif.fr,
laura.toni@iit.it

Abstract

Due to its applications in marine research, oceanographic, and undersea exploration, Autonomous Underwater Vehicles (AUVs) and the related control algorithms has been recently under intense investigation. In this work, we address target detection and tracking issues, proposing a control strategy which is able to benefit from the cooperation among robots within the fleet. In particular, we introduce a behavior-based planner for cooperative AUVs, proposing an algorithm able to search and recognize targets, in both static and dynamic scenarios. With no a priori information about the surrounding environment, robots cover an unknown area with the goal of finding objects of interest. When a target is found, the AUVs' goal become to classify it (fixed target) or track it (mobile target), with no information about target trajectory and with the constraint on maintaining the formation. Results demonstrate the good overall performance of the proposed algorithm in both scenarios.

Index Terms

Mission planner, Behavior-based methods, Artificial Potential Fields, Autonomous Underwater Vehicle, target detection and analysis.

[†] Corresponding author.

Part of this work has been presented to OCEANS '11 MTS/IEEE Conference.

I. INTRODUCTION

In recent years, the marine robotic research community has been increasingly attracted by the design of cognitive systems for coordination and cooperative control of multiple underwater vehicles. This interest has been driven by the high improvement a cooperative strategy can provide in several activities, such as monitoring marine habitat, collecting underwater data, or exploring areas difficult or impossible for human to reach. The benefits of exploiting group of robots as opposed to single become evident when considering performances, costs, fault tolerance and reconfigurability [10]. In this framework, several improvements have been achieved in the AUVs technology, due to its potentiality in several underwater applications (e.g., Mine-like Objects (MLOs) classification [6], Magnetic Field Measurement [18]), turning the idea of employing multiple AUVs on a coordinated mission into a realistic goal. In the underwater scenario, most of the efforts have been focused on developing and studying algorithms for the detection and recognition of underwater targets [27], involving both discrimination between targets and non targets and their classification. Within this framework, researchers have been addressing the cooperation among multiple AUVs, which might cover an area wide far beyond the possibilities of a single vehicle or the same area of a single robot in a reduced amount of time, leading to an improvement of the task accomplish rate and, at the same time, reducing the error probability in classifying the object.

When a cooperative mission for target detection and recognition is considered, the task might be classified based on whether the targets are *fixed* or *mobile*. In literature some works focus on the detection and classification of fixed targets, as in [21], [12] in opposition to others focused on tracking of mobile objects, as in [25]. As long as fixed targets are concerned, the cooperation

might lead the team to assume an opportunistic displacement around the target, augmenting the precision level in its detection and classification [9]. Besides, for mobile targets, the fleet of AUVs might follow the object in order to track its behavior (i.e. marine mammals) and use various sensors to track also the object-environment interaction. It follows that, in mobile target scenarios, the fleet is employed in a mission planner which dynamically changes, based on the target movement and the variation of the surrounding background. This arises the need of developing an algorithm able to learn and to correctly react to the changing scenario.

To control Multiple Autonomous Underwater Vehicles (MAUVs) employed to localize targets in an unknown environment, a Behavior-Based Method (BBM) has been introduced in [24]. Several BBMs have been developed for mobile robot applications, demonstrating their efficiency in controlling robots having no knowledge about the area to explore [2], [20]. Applied to the underwater world, BB techniques for target detection with a single AUV have been proposed in [5]; while, in [20], authors employed behavior rules to control a predefined formation of MAUVs. A path planning control strategy for searching and classifying task using multiple cooperative Underwater Vehicles has been investigated in [9], where the authors exploit the diversity offered by the fleet, assigning different roles to different robots.

One of the most common implementation for this kind of missions uses Artificial Potential Fields (APFs) [15]: robots are treated as a particle under the influence of a potential field, whose local variation is related to the sensed environment. In [26], for example, Yang et al. used APFs to coordinate three kinds of potentials: an attractive one to the target, a repulsive one from obstacles, and a third interactive potential among robots. This attracting and reacting potentials should drive each robot of the fleet to take the right action, based on the sensed

environment and the mission goal. However, problems linked with the APF methods like local minima, or constraints due to AUVs dynamics have been neglected, while a priori information about the target position have been considered as a starting input of the algorithm.

In our previous work [24], a Behavior-Based (BB) approach was considered to control multiple AUVs aimed at detecting and localizing fixed targets with no a priori information about the environment.

In this paper, we aim at improving our previous work, proposing a complete mission planner for a cooperative fleet of AUVs, employed in a mission of target detection and tracking, in case of both static and dynamic targets. To reach this goal, the proposed algorithm needs to include formation issues and a new type of control technique to reach the target position. In particular, we propose a mission planner aimed at localizing and recognizing fixed objects of interest, using a BB controller together with a real-time APF controller. The former is used for area coverage with the goal of target localization, while the latter, which allows the fleet to opportunistically displace itself around the target, is considered for target recognition. We also demonstrate the flexibility of the proposed mission planner in a dynamic scenario with mobile targets. Results show that our algorithm is able to opportunistically adapt to a scenario that dynamically changes, without having a priori knowledge about the environment and targets.

The main contributions of our work are the following:

- integration of different phases of a mission in order to recognize objects of interest;
- integration of various techniques to face the main problems related to APFs, such as local minima;
- study and implementation of a new interaction force between robots, based on a modified

Van der Pol oscillator;

- development of an novel APF controller;
- application of APF forces to a dynamic AUVs model, by numeric integration of force.

The reminder of this paper is organized as follows: in Section II, we describe the key concepts of the Behavior-Based systems. In Section III, we describe the global structure of the proposed mission planner; while, in Section IV, we briefly remind the BB algorithm presented in [24], and the relative results for a target detection mission. Section V introduces the APF controller developed to head robots towards a goal position, represented by a detected target. Then, in section VI our simulation tools is detailed, with the focus on the realistic AUV model chosen to validate our technique. Section VII depicts simulation results and analyzes performances of the algorithm in several scenarios; the paper concludes with a summary of the results and suggestions for further research.

II. THE BB SYSTEMS

The origins of Behavior-Based systems have to be searched in the animals' world, where each subject is capable of quickly adapt to changing conditions by using few simple behaviors based on the perceived environment. This principle can be applied to BB control, where a set of distributed, interacting modules, called *behaviors*, are employed in order to achieve a desired emerging behavior of the system [16]. Behaviors are designed at different levels, starting from the survival behaviors (e.g. obstacle avoidance) to more complex ones that could be also a subsystem of combined elementary behaviors. Thus, the BB controller is similar to a structured

network of interacting agents with no centralized world representation or supervisor. These features let BB systems be easy to implement and highly adaptive to operative conditions changes, leading the BB techniques to be extremely suitable for robot controllers.

BB techniques were originally developed for single robot systems, and then extended to a multi-robot scenario. Even within a team, the motion of each vehicle is generated by a fast behavior based only on its current sensory information, typically through a very simple sensor-motor mapping [2]. Thus, several simple low-level behaviors (e.g., “avoid obstacle”) can be combined between each other, to obtain higher-level responses (e.g., “path planning”), leading to a global emerging behavior. It is worth noting that, since BB systems were originally inspired by reactive systems and, since both of them maintain the real-time coupling sensing-acting, BB techniques are usually equated to reactive systems. Actually, they subtly yet substantially differentiate in internal states definitions. Reactive systems are not capable of using internal state representation and learning, while BB systems, thanks to the behavior, can store internal state representation in a distributed fashion. In other words, the reactive control follows the rule *Sense, don't think and (re)act*, while BB approach bases its theory on *thinking on the way of acting* before any action. This difference makes BB systems more powerful than reactive ones, and it is one of the main reasons for the BB flexibility [16]. As shown in [16], the key concepts of BB controls are:

- Implementation as control laws defined as a processing element of procedure;
- Behaviors are designed to be simple and easy to implement;
- Each behavior can take inputs from sensors or other modules and send outputs to the system actuation module;
- Different behaviors can take inputs from the same sensors and give outputs to the same

actuators;

- Behaviors are executed concurrently in order to exploit parallelism and speed of computation.

A well known BB architecture has been developed by Arkin in [2], where the author proposed the Motor-Schema (M-S) approach. Based on the reaction to different sensory inputs, each individual behavior can be combined achieving a resultant control vector that takes into account all the simple behaviors. Assuming that the sensor-motor reactions can be written as simple velocity control vectors, the resultant vector can be evaluated as a vector sum. One of the most common implementation of the M-S approach uses APFs.

In the early days of robot motion planning, APFs arose as a simple, computationally efficient planning routine. To draw or repel an object (the robot) from one point to another one, APF methods use a simplified version of laws of nature, (i.e., attractive and repulsive potential), as suggested by Khatib in [15], where the concept of imaginary forces field driving robot was introduced. In M-S techniques, an attractive or repulsive potential constitutes the emerging behavior that arises as reaction to a sensory stimulus. A wide variety of behavior commands can be derived, such as “Avoid Obstacle”, “Move to Light” and “Random Search” [3].

APF theory uses the positions of two points of interest to calculate attractive and repulsive vectors. The potentials are based on the vector connecting the two points. The nature of the resulting potential will depend on the underlying behavior. Figure 1 shows an example of how the attractive and repulsive vectors are summed to produce a resultant vector that drives the robot from a start point to a target.

In summary, the power and complexity of BB systems take origin from the way each

behavior is designed and from how they are coordinated to obtain a desired emerging behavior.

III. PROPOSED MISSION PLANNER

Motivated by several applications introduced in the previous sections, the aim of this work is to study and develop a complete mission planner, controlling a group of AUVs deployed in an unknown a priori underwater environment. The main goal of the fleet is to cover a given area in order to search for objects of interest, which can be static (i.e., archaeological finds, coral trees) or moving (i.e. marine mammals, other marine vehicles); once found, these objects become targets to be analyzed from the whole fleet in order to be recognized. This mission goal leads to decompose the global task in two fundamental steps:

- 1) Target detection phase;
- 2) Target analysis phase.

In the first phase, depicted in Fig. 2(a), AUVs need to cover an unknown area in the most efficient way, with the final goal of finding targets while avoiding obstacles and other robots on their paths. The aim is to achieve a global emerging behavior of the fleet in which each AUV is responsible for taking decisions about its own motion and without explicit inter-robot coordination. These specifications can be easily hold by a BB approach, as the one introduced in [24] and described in the next section, where the effectiveness of this technique is demonstrated when robots have to spread in an unknown environment.

When an interesting object is detected, it becomes a target and the second phase starts (see Fig. 2(b)). The aim is to analyze the target, in such a way that each member of the fleet contributes

by giving a different point of view of the object. In order to allow robots to accomplish this task, we propose an APF controller heading robots towards the object to be analyzed, while overcoming obstacles, keeping formation of AUVs in desired geometry and avoiding collisions between them. A main objective of this approach is to implement distributed control and eliminate the ordering of the AUVs. A detailed description of the APF controller components is described in section V.

IV. PROPOSED BB APPROACH FOR TARGET DETECTION

We now present the algorithm developed for area coverage (Fig. 3), able to find object of interest based on a BB approach, which is the goal of the first phase of the mission planner. With this aim, we develop a BB technique, to control a fleet of cooperating AUVs during a target detection task. The scenario is based on an unknown underwater environment with fixed obstacles and targets, where each robot has to move at a constant depth. A 2D scenario is typically considered in a large number of missions, as described in [7]; for this reason, we limit our model to work in a 2D underwater scenario. In the following, dynamics of a real AUV will be disregarded, and the study will be focused on kinematic simulations, which are extremely helpful in verifying the method efficiency; the test on a realistic AUV model, characterized by nonlinear dynamics, will be presented in the following section.

We suppose that AUVs are equipped with high-quality positioning and sonar systems which provide, as in [17], reliable self-positioning and location information about obstacles, targets and robots within the sensor range. In the proposed algorithm, APFs [15], [28] are considered to

represent the relationship between the robot and the environment; in particular the APF forces are evaluated as Social Potential Fields defined in [22], where the interaction between two robots is given by:

$$\mathbf{f}(d) = \left(-\frac{c_1}{d^{\alpha_1}} + \frac{c_2}{d^{\alpha_2}} \right) \mathbf{u}_d \quad (1)$$

where $\mathbf{u}_d$ is the versor and d is the distance between the two robots; while, $c_1, c_2 \geq 0$, $\alpha_1, \alpha_2 > 0$ are parameters to be set. The resulting force in (1) is adopted to describe the interaction not only between two AUVs but also between the robot and the surrounding environment. Note that setting c_1 equal to zero means a $\mathbf{f}(d)$ purely attractive; viceversa setting c_2 equal to zero leads $\mathbf{f}(d)$ to be a purely repulsive force. During the mission, each vehicle carries and updates two different grid-based maps: a Local Map (LM) and a Global Map (GM). The choice of grid-based maps has been driven by their simplicity, flexibility and low computational requirements [5], important features for the key aspect of energy consumption in underwater missions. As depicted in Fig. 4, the LM contains information about the present sensed area, discretized into $m \times m$ cells; the GM, instead, represents the entire environment under investigation and stores values of different LM until a given iteration. At each time step, APF forces resulting from the available sensor information are evaluated in correspondence of the central position of each LM cell, and the relative intensity values are stored both in LM and in the corresponding cells of GM. In this way, the GM keeps the global knowledge of the environment acquired during the mission until the current iteration, and can be used to help the robot exploring areas already unknown. Values from both the maps are then used to determine the desired robot next position, instead of the robot speed as in classical BB techniques. The main steps of the

algorithm (see Fig. 3) are the following:

- **Initialization;**
- **Step 1. GM sharing:** if possible, AUVs share their GM, merging information already acquired;
- **Step 2. GM evaluation:** the AUV uses the global information acquired to direct its movement toward the targets or uncovered areas;
- **Step 3. LM evaluation:** the AUV evaluates the APF forces generated by what they can currently sense;
- **Step 4. LM resulting force:** the overall force resulting from the evaluation of the LM is evaluated, together with an additional random force used to avoid local minima;
- **Step 5. Next movement:** each robot computes the overall force resulting from both GM and LM, obtaining the next position to reach.

Initialization We suppose to have r robots cooperating in a target detection mission. As introduced above, each AUV maintains a GM of $M \times M$ cells, where $n_{GM} = M^2$ is the total number of squared cells in which the area to explore is divided. This map is filled with the force values computed in relation to the area under investigation. On the other hand, the LM, which represents the sensed area, is split into $n_{LM} = m \times m$ squared cells. The cells are indexed following a raster scan both in the GM and LM, and each cell has dimensions $l \times l$ meters. When the mission starts, all the AUVs employed will be distributed close one another in a small area randomly chosen in the GM. Then, the high level controller on each vehicle evaluates the next movement following Steps (1-5) for each j -th iteration, where $j=1, \dots, T_t$, and T_t is the number of iterations required to accomplish the task.

Step 1. GM sharing In this step, the information sharing among AUVs is considered. We will suppose that two AUVs can share information when their distance d_i is below a given threshold d_{th} , otherwise the algorithm jumps to the next step. The communication ranges considered in the following are evaluated taking into account realistic acoustic modem, bandwidth and transmission parameters [8]. Since the evaluation of the communication effects on the BBM algorithm is out of the scope of this paper, here we assume ideal transmission (i.e., with no channel impairments during the transmission). A study of realistic transmission affecting the considered model will be a future work. During the communication, vehicles merge their information about the discovered environment by exchanging their GMs. Each AUV refreshes the cells of their GM, as described in the following formula:

$$GM_i(j) = \text{sign}(GM_i^1(j) + GM_i^2(j)) \cdot \max(|GM_i^1(j)|, |GM_i^2(j)|) \quad (2)$$

where GM_i^1 (GM_i^2) is the i -th cell of the AUV 1 (AUV 2) and , the subscript i is referred to a raster scan indexing. This formula guarantees that each AUV GM is updated with the latest available information; as a consequence areas already covered by the first AUV will not be explored a second time by another AUV during target searching.

Step 2. GM evaluation The vehicle is able to store information acquired up to iteration j , so it can use GM to direct its general movement. Since we consider a static environment, each vehicle is subject to an attractive force toward uncovered areas. At the j -th iteration, the overall force applied to one robot by its unexplored regions is evaluated based on (1) as follows:

$$\mathbf{f}_{total}^{GM}(j) = A_k^2 \sum_{i=1}^k \frac{\mathbf{u}_d}{d_i^3} \quad (3)$$

where A_k represents the ratio between covered ($n_{GM} - k$) and uncovered (k) cells at iteration j , $\mathbf{u}_i$ is a versor from the AUV position pointing to the i -th cell, and d_i is the relative distance between them.

Step 3. LM evaluation To avoid obstacles or to reach targets within the AUV sensor range, the next robot movement will be mainly influenced by the last sensed information. Thus, at the j -th iteration, each robot updates its LM merging its current sensory information with previous knowledge stored in its GM. Each i -th cell of the LM will have its own potential field value, and the relative force is given by:

$$\mathbf{f}_i(j) = \left(GM_i(j) + \frac{c_{gain}}{d_i^2} \right) \mathbf{u}_i \quad (4)$$

where d_i is the distance between the robot, $\mathbf{u}_i$ is the relative versor and the considered cell in the LM and c_{gain} is a setting parameter; obstacles and visited areas will generate a repulsive force ($c_{gain} < 0$), while targets will produce an attractive force over the AUV ($c_{gain} > 0$). Other cells will not affect the forces ($c_{gain} = 0$), and the values obtained are stored also in the relative cells of GM.

Step 4. LM resulting force In this step, the overall force acting on the vehicle due to information of the LM is evaluated as follows:

$$\mathbf{f}_{total}^{LM}(j) = \sum_{i=1}^{n_{LM}} \mathbf{f}_i(j) + \mathbf{f}_{rand} \quad (5)$$

$\mathbf{f}_{rand}$ is a random vector used to avoid local minima, typically arising in potential field base solutions [2].

Step 5. Next movement Finally, the robot computes the total force as:

$$\mathbf{f}_{total}(j) = \mathbf{f}_{total}^{GM}(j) + \mathbf{f}_{total}^{LM}(j) \quad (6)$$

The vector $\mathbf{f}_{total}$ represents the desired heading, and we assume that the robot movement will be limited by the LM boundaries, ensuring the avoidance of obstacles or other robots. At the end, values of the current LM are stored into the corresponding cells of GM, in order to maintain the mission history and to be used in next algorithm iterations. Fig. 4 shows a schematic representation of the considered scenario.

V. MODIFIED APF CONTROLLER FOR TARGET RECOGNITION

In this section, we describe the technique studied for the second phase of a mission. We assume that robots are not supposed to know whether the target is fixed or mobile. We developed a novel APF controller that heads robots from their current position toward the target in both cases. As previously mentioned, when an AUV discovers a target, the first phase ends and the second begins. In this part of the mission, the other robots have to reach the target maintaining formation constraints. The APF controller is a real-time path-planner that guarantees to reach a target positions, avoiding obstacles and other robots while preserving formation constraints among AUVs. We will see that the same APF controller is perfectly suitable if the target position is not a point but a trajectory. It is worth underlying that there is no central supervision, but each AUV is guided by its own APF controller, leading to an emerging behavior of the fleet.

The total force derived from the potential field of each vehicle is evaluated as the sum of:

- an attractive force to the target, as described in [26];
- a repulsive force from the obstacles that are within a sensor range;
- a force that describes the interaction between robots based on a modified Van der Pol oscillator [13].

The main force acting on the robot is the result of a new APF based on [26], defining an attractive and repulsive force on the basis of the sensed environment in an innovative way. To avoid local minima and obstacle proximity, two additional techniques have been introduced. To face the first problem we consider rotors, vectorized quantities which introduce directionality, and thereby cause divergence from local minima, as described in [4]; they have proved to be very efficient in this kind of scenarios where no a priori knowledge is available. In order to guarantee the obstacles avoidance when robots enter in a too close area around them, the force controlling robots movement is forced to be only a repulsive one from the obstacle, heading the robot away and preserving its safety.

Focusing on the third type of force, to keep formation of AUVs in desired geometry and avoid collisions between them, we present a new approach based on the Van der Pol oscillator [13]. It is a mass-spring-dumper system characterized by a non-linear damping coefficient. The classical Van der Pol oscillator is defined by the following equation:

$$\ddot{x}(t) = \frac{k}{m}x(t) - \frac{f}{m}(x^2(t) - 1)\dot{x}(t) \quad (7)$$

It describes the linear displacement of the mass m under the influence of a spring, with elastic coefficient k , and a dumper with coefficient $f(x^2(t) - 1)$. The last factor guarantees that the spring and dumper deformations do not grow indefinitely nor become zero (for more details we refer the reader to [13]). We modified the equation (7) on the basis of our formation

constraints:

- the AUVs must be under a maximum relative distance R ;
- the AUVs must be over a minimum security relative distance d_0 .

The first constraint modifies the spring factor, while the second one acts on the damper coefficient. It is worth noting that in our scenario we have to adapt the equation (7) for 2D movements. The resulting modified equation is:

$$\mathbf{F}_{tot} = \mathbf{F}_{att} + \mathbf{F}_{rep} = k(d - R)\mathbf{e}_d - f\left(\frac{d_0}{d}\right)\mathbf{v} \quad (8)$$

where d is the euclidean distance between robots, $\mathbf{e}_d$ is the versor of the line connecting the two robots and $\mathbf{v}$ is the velocity vector of the AUV under investigation. Figure 5 shows a schematic representation of this approach; figure 6 represents the trend of the force deriving from the modified Van der Pol oscillator, acting on a material point with unit mass with respect to a fixed point. It is worth noting that, with opportune parameters calibration, the position of the material point tends to R and, at the same time, it is never below the minimum distance d_0 . The forces acting on robots during navigation are numerically integrated to obtain the reference velocity for each vehicle. The calibration of these forces is not trivial and defines the precision and efficiency of the controller. Besides, problems can arise from the force integration, computed in order to obtain reference velocities, because of system inertia and dynamics.

VI. AUV MODEL AND CONTROL

In this section, we briefly introduce our simulation tool developed to test and analyze the whole mission planner. Our approach has been implemented in Matlab Simulink, where a real AUV has been modeled too, in order to study the feasibility of the presented technique in a more

realistic way. We considered the REMUS AUV model and its relative dynamic parameters as in [19]. This is an underactuated vehicle equipped with two control surfaces, one for diving and one for steering, and a thruster for the forward motion. The REMUS, together with other class of AUVs, can be represented as a six degree-of-freedom rigid body subject to external forces, governed by the following equations:

$$\begin{aligned}\mathbf{M}\dot{\mathbf{v}} + \mathbf{C}(\mathbf{v})\mathbf{v} + \mathbf{D}(\mathbf{v})\mathbf{v} + \mathbf{g}(\boldsymbol{\eta}) &= \boldsymbol{\tau} \\ \dot{\boldsymbol{\eta}} &= \mathbf{J}(\boldsymbol{\eta})\mathbf{v}\end{aligned}\tag{9}$$

where $\boldsymbol{\eta} = [x \ y \ z \ \phi \ \theta \ \psi]'$ is the vector of linear and angular position relative to the Earth-Fixed frame, $\mathbf{v} = [u \ v \ w \ p \ q \ r]'$ is the vector of linear and angular velocities in the Body-Fixed frame; $\mathbf{J}(\boldsymbol{\eta})$ is a rotation matrix, $\mathbf{M}$ is the mass matrix, sum of rigid body mass and added mass matrices, $\mathbf{C}(\mathbf{v})$ is the Coriolis matrix due to rigid body and added mass, $\mathbf{D}(\mathbf{v})$ is the total damping matrix, $\mathbf{g}(\boldsymbol{\eta})$ is the vector of hydrostatic forces and $\boldsymbol{\tau}$ is the vector of external forces generated by the actuation system. In the particular case of the REMUS, $\boldsymbol{\tau}$ is function of the propeller revolute-per-minute (rpm) ($n(t)$), the deflection of rudder ($\delta_r(t)$) and the deflection of stern ($\delta_s(t)$) (for more details about modeling the reader is refereed to [19] and [11]). Based on [14] and [11], we have implemented the model described in MATLAB Simulink and we have designed a motion controller with a sliding-mode approach. The basic idea is to define a sliding surface $\sigma(t)$ as a weighted difference between real measurements and desired outputs. The resultant control law needs to be able to “slide” system trajectories on $\sigma(t)$ by the use of a positive constant μ and an intermittent signal. Besides, a proper tuning of μ guarantees the stability of the system and the robustness of the controller. As described in [14] and [11], we design a different law for each control input $n(t)$, $\delta_r(t)$ and $\delta_s(t)$, obtaining:

Forward Speed Control Let $u(t)$ be the forward speed, the signed square of the rpm $n(t)$ of the thruster is:

$$n(t)|n(t)| = \frac{m - X_{\dot{u}}}{f_p} \left(-\frac{X_{uu}}{m - X_{\dot{u}}} u(t)|u(t)| - \mu_1 \tanh\left(\frac{u(t) - u_d(t)}{\varphi_1}\right) + \dot{u}_d(t) \right) \quad (10)$$

where f_p is a propeller factor depending on the fan diameter and water density, $X_{\dot{u}}$ and X_{uu} are model parameters, m is the mass of the vehicle, φ_1 is the “boundary layer thickness” that solves the chattering problem and μ_1 is the positive constant introduced above.

Steering Control The variables of interest in the steering motion are the sway velocity $v(t)$, the yaw $r(t)$ and the angle $\psi(t)$. The control input is the rudder deflection $\delta_r(t)$. The sliding surface is defined as:

$$\sigma_2(t) = \mathbf{h}_1(1)(v(t) - v_d(t)) + \mathbf{h}_1(2)(r(t) - r_d(t)) + \mathbf{h}_1(3)(\psi(t) - \psi_d(t)) \quad (11)$$

To stabilize the linearized subsystem we chose $[-0.45 \ -0.46 \ 0]$ as closed-loop poles, obtaining a linear feedback $\mathbf{k}_1 = [4.6840 \ 1.0518 \ 0]$. The weights in $\mathbf{h}_1$ are computed as the right eigenvector of the eigenvalue equal to zero, obtaining $\mathbf{h}_1 = [0.81845 \ 0.571940 \ 0.54983]'$. Finally the control law is the following:

$$\begin{aligned} \delta_r(t) = & -\mathbf{k}_1(1)v(t) - \mathbf{k}_1(2)r(t) + \frac{1}{\mathbf{h}_1' \cdot \mathbf{b}_1} (\mathbf{h}_1(1)\dot{v}_d(t) + \\ & + \mathbf{h}_1(2)\dot{r}_d(t) + \mathbf{h}_1(3)r_d(t) - \mu_2 \tanh\left(\frac{\sigma_2(t)}{\varphi_2}\right)) \end{aligned} \quad (12)$$

where $\mathbf{b}_1$ is the input matrix, that here is a vector, of the linearized subsystem, $\mu_2 = 1$ and $\varphi_2 = 0.8$.

Diving control In this case the variables of interest are the position $z(t)$, the heave $w(t)$, the

pitch $q(t)$ and the angle $\theta(t)$. Following the same procedure, we defined the sliding surface:

$$\begin{aligned}\sigma_3(t) = & \mathbf{h}_2(1)(w(t) - w_d(t)) + \mathbf{h}_2(2)(q(t) - q_d(t)) + \\ & + \mathbf{h}_2(3)(\theta(t) - \theta_d(t)) + \mathbf{h}_2(4)(z(t) - z_d(t))\end{aligned}\quad (13)$$

Choosing $[-0.35 \ -0.36 \ 0 \ -0.37]$ as closed-loop poles, we obtained $\mathbf{k}_2 = [-4.4990 \ -0.9068 \ 0 \ -0.0187]$, with a resultant $\mathbf{h}_2 = [0.7943 \ 0.5975 \ 0.1095 \ 0.0091]'$. The control law is the following:

$$\begin{aligned}\delta_s(t) = & -\mathbf{k}_2(1)w(t) - \mathbf{k}_2(2)q(t) - \mathbf{k}_2(4)z(t) + \frac{1}{\mathbf{h}_2' \cdot \mathbf{b}_2} (\mathbf{h}_2(1)\dot{w}_d(t) + \\ & + \mathbf{h}_2(2)\dot{q}_d(t) + \mathbf{h}_2(3)\dot{\theta}_d(t) + \mathbf{h}_2(4)\dot{z}_d(t) - \mu_3 \tanh\left(\frac{\sigma_3(t)}{\varphi_3}\right))\end{aligned}\quad (14)$$

where $\mathbf{b}_2$ is the input matrix of this system, the parameters μ_3 and φ_3 are 1 and 0.1 respectively. This controller supposes that it is possible to obtain some velocities values from the desired path/trajectory. It is worth noting that, focusing on 6 DOF systems for a 2D application is useful to verify the guidance performances in a realistic scenario.

The resulting sliding mode controller of this procedure has a problem related with the steady-state error. In fact, as illustrated in [23] the heading angle ψ presents a large steady-state error, due to the fact that the sliding mode steering controller is a nonlinear controller and there is no integrator action like the conventional PID controller to eliminate this error. We observed, from simulations, that this is a problem that affects all the controllers, making them slow compared with the dynamics of the system.

To overcome these problems, according also with [23], we implemented a PI action for each controller that is added at the control action of the main sliding-mode controller. This leads to

obtain a more reactive and precise controller with no steady-state error.

Concerning the guidance law, in our application, in the first phase of the mission, the vehicle has to follow waypoints specified real-time by the high-level controller, while, in the second phase, it has to follow reference signals regarding the desired position and attitude. During the first phase, the controller guides the AUV to reach waypoints, steering with a desired heading angle while retaining a given altitude, and without time constraints. This kind of autonomous guidance is most simply accomplished by approaching the Line Of Sight (LOS) between the vehicle present position and the waypoint to be reached. We implemented a LOS technique as described in [14]. We refer the reader to [1] for a solid theoretical background on the problem. In short, a LOS method consists mainly in calculating the heading angle $\psi_d(t)$ step by step considering the desired waypoint and the current position, through the formula:

$$\psi_d(t) = \arctan\left(\frac{y_d - y(t)}{x_d - x(t)}\right) \quad (15)$$

where $x(t)$ and $y(t)$ correspond to the position of the vehicle at time t . Regarding $u_d(t)$, we consider a constant velocity equal to the nominal speed $u_0 = 1.5 \text{ m/s}$. The waypoint switching logic changes to next point, computed by the high-level controller, when the AUV is positioned in a sphere of acceptance of radius ρ_0 of the current waypoint [14]. Figure 7 shows the x and y coordinates of the robot path, obtained with 16 waypoints and $\rho_0 = 2 \text{ m}$, that is the length of the vehicle. In the second phase, the vehicle simply follows reference signals and the guidance law is the output of the described APF controller.

VII. RESULTS

We now show results carried out with our MATLAB/Simulink simulator for scenarios

characterized by fixed obstacle and fixed/mobile targets, always with a fleet of three AUVs. The dimensions of the area under investigation can vary from 70 by 20 m to 1 by 1 km. In the following, we first provide results for static scenario, and then we demonstrate the validity of the proposed algorithm even in dynamic conditions. Fig. 8 depicts the result of a complete mission accomplished by three AUVs in a static scenario with a fixed obstacle: the continuous line represents the first phase of the mission (target detection), while the dotted one represents the second phase (target recognition) that starts from the indicated points (dots). We remind that, the second part of the mission starts when a robot discovers a target. During the second phase, led by the discovering robot that remains close to the target, the rest of the fleet will have to reach a position around the discovered target, sorting in asymmetric pattern to analyze and recognize the target. So, in our simulated scenario, only two AUVs (AUV 1 and 3) out of three will move during the second phase of the mission (the leader robot will be mainly in a fixed position). The stars highlight the goal positions of each robot while squares and circles are obstacles. This figure shows that a mission is completed in a safe and efficient way, and the ability of AUVs to avoid obstacles and other robots. It is worth noting that formation constraints lead to a path that is not a straight line from the position to the target, but it is slightly curved.

To better understand the behavior of formation constraints, in Fig. 9 we show the result of the aforementioned Van der Pol oscillator relation applied to a fleet of three AUVs. The red rectangle represents an obstacle through the way to the target positions; the circles indicate temporally subsequent robot positions in different steps of the algorithm. During the first steps of the mission the initial condition (the spring rest condition) is preserved in the free space.

Then, even if the formation is perturbed during obstacle avoidance, goal positions are reached while keeping safety distance between vehicles and avoiding the obstacle.

We now consider the dynamic scenarios, in which targets are moving. Fig. 10(a) shows the result of a tracking of a target that follows an elliptical path. It can be noticed that all the robots follows the target maintaining always formation constraints, guaranteeing collision avoidance between them and obstacles. The obstacle avoidance behavior can be observed in the middle of the path (X Position in the range [440,500]m), is caused by obstacle proximity. However, even if the perturbed state, the fleet is able to keep the formation and reach the goal. The good performance of the tracking is also confirmed by Fig. 10(b), where the error distance between the target and the first robot is shown. It is worth nothing that the distance is always below a given value (2m); besides after perturbation due to the obstacle proximity (between 200s and 300s) the fleet reaches rapidly the previous state and tracks the target.

In Fig. 11, we provide the results for a simulated mission planner, in which the target moves following a sinusoidal path, which results to be a more complex path (compared to the elliptical one) due to the several changes of direction. Fig. 11(a) depicts the result of the mobile target tracking; here the tracking is not as precise as in the previous case, and this is justified by the fact that the path of the target is more variable and the fleet bases its movement only on the effective position of the target and the sensed environment. However, despite the less precision in keeping the formation, the fleet is still able to track target during its movement. This is confirmed by Fig. 11(b) that illustrates the distance error between the first AUV and the target: even if it is bigger than the other one, the error remains always limited.

VIII. CONCLUSIONS

We presented a novel Mission Planner for target searching and recognition with Autonomous underwater Vehicles (AUVs). The mission is divided in two phases, each one accomplished by a Behavior-Based (BB) technique: 1) target detection, 2) target analysis. During the searching task, an approach based on a BB system combined with Grid-Based Maps is applied, while, in the second phase, we propose an approach still based on BB techniques but with a real-time Artificial Potential Field (APF) controller that drives each robot toward the goal position. With this proposed Mission Planner, robots are able to find targets in an efficient way, avoiding obstacles and intra-fleet collisions. Once the target is found, the fleet is able to reach it, being it fixed or moving. This is accomplished with no a priori knowledge and keeping formation constraints based on a modified Van der Pol oscillator equation.

Results confirmed the validity of this Mission Planner in both static and dynamic scenarios, underlying the good behavior of the fleet with the tracking of a mobile target.

REFERENCES

- [1] A.P. Aguiar and A.M. Pascoal. feb. 2007. Dynamic positioning and way-point tracking of underactuated auvs in the presence of ocean currents. *International Journal of Control*.
- [2] R. C. Arkin, 1998. *Behavior-based robotics*. MIT Press.
- [3] T. Balch and R.C. Arkin. 1998. Behavior-based formation control for multirobot teams. *IEEE Transactions on Robotics and Automation*, 14(6):926–939.
- [4] M. Barisic, Z. Vukic, and N. Miskovic, 2008. Effects of rotors on uuv trajectory planning via the virtual potentials method. In *16th Mediterranean Conference on Control and Automation*.
- [5] A. Bennett and J.J. Leonard. apr 2000. A behavior-based approach to adaptive feature detection and following with autonomous underwater vehicles. *IEEE Journal of Oceanic Engineering*, 25(2):213–226.
- [6] B.Johnson, N. Hallin, H. Leidenfrost, M. O’Rourke, and D. Edwards. 2009. Collaborative mapping with autonomous underwater vehicles in low-bandwidth conditions. In *OCEANS 2009*, pages 1–7.
- [7] E. Bovio, D. Cecchi, and F. Baralli. 2006. Autonomous underwater vehicles for scientific and naval operations. *Annual Reviews in Control*, 30(2):117 – 130.

- [8] S.S.M. Chitre and M. Stojanovic. 2008. Underwater acoustic communications and networking: Recent advances and future challenges. *Marine Technology Society Journal*, 42:103–116.
- [9] Y. Deng, P.-P.J. Beaujean, E. An, and E.A. Carlson. 2008. A path planning control strategy for search-classify task using multiple cooperative underwater vehicles. In *OCEANS 2008*, pages 1–9, 15-18.
- [10] G. Dudek, M. Jenkin, E. Milios, and D. Wilkes. jul 1993. A taxonomy for swarm robots. In *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, pages 441–447.
- [11] T. I. Fossen., 1994. *Guidance and Control of Marine Vehicles*. John Wiley, and Sons
- [12] L. Freitag, M. Grund, C. von Alt, R. Stokey, and T. Austin. 2005. A shallow water acoustic network for mine countermeasures operations with autonomous underwater vehicles. In *Underwater Defense Technology UDT*.
- [13] Alessandro Giua and Carla Seatzu. 2009. *Analisi dei sistemi dinamici*. Springer Milan.
- [14] A.J. Healey and D. Lienard. 1993. Multivariable sliding mode control for autonomous diving and steering of unmanned underwater vehicles. *IEEE Int. Journal of Oceanic Engineering*.
- [15] O. Khatib. mar 1985. Real-time obstacle avoidance for manipulators and mobile robots. In *Proc. of IEEE International Conference on Robotics and Automation*, volume 2, pages 500–505.
- [16] Maja J. Mataric and François Michaud. 2008. Behavior-based systems. In *Springer Handbook of Robotics*, pages 891–909. Springer Berlin Heidelberg.
- [17] J.B. Paduan, D. W. Caress, D. A. Clague, C. K. Paull, and H. Thomas. 2009. High-resolution mapping of mass wasting, tectonic, and volcanic hazards using the mbari mapping auv. *Int. Conference on Seafloor Mapping for Geohazard Assessment*.
- [18] J. Pentzer, B. Crosbie, T. Bean, J. Canning, J. Frenzel, M. Anderson, and D. Edwards. may 2010. Measurement of magnetic field using collaborative auvs. In *OCEANS 2010 IEEE – Sydney*.
- [19] T. Presterio. 2001. Verification of a six-degree of freedom simulation model for the remus autonomous underwater vehicle, master thesis, MIT/WHOI, 129 pp.
- [20] Q. Jia and G. Li. aug 2007. Formation control and obstacle avoidance algorithm of multiple autonomous underwater vehicles (AUVs) based on potential function and behavior rules. In *IEEE International Conference on Automation and Logistics*.
- [21] A. Rajala and D. Edwards. 2006. Allocating auvs for mine map development in mcm. In *OCEANS 2006 - Asia Pacific*, pages 1–8, 16-19.
- [22] J. Reif and H. Wang. 1995. Social potential fields: A distributed behavioral control for autonomous robots. In *Robotics and Autonomous Systems*, pages 171–194.
- [23] J. Shi. nov. 2006. Design of sliding mode autopilot with steady-state error elimination for autonomous underwater vehicles. In *TENCON 2006. 2006 IEEE Region 10 Conference*.
- [24] L. Sorbi, G.P. De Capua, J. Fontaine, L. Toni, and L. Rossi. 2011. Behavior based control: An approach to cooperative AUVs missions. In *The 16th IASTED International Conference on Robotics and Applications*.
- [25] L. Sorbi, L. Toni, G.P. De Capua, and L. Rossi. dec. 2010. Communication in a behavior-based approach to target detection and tracking with autonomous underwater vehicles. In *2010 IEEE/IFIP 8th International Conference on Embedded and Ubiquitous Computing (EUC)*, pages 320 –325.
- [26] Y. Yang, S.X. Wang, and Z.L. Wu. 2010. Multi-auv coordination in the underwater environment with obstacles. In *OCEANS 2010 IEEE – Sydney*.
- [27] C. Yuan, M.R. Azimi-Sadjadi, J. Wilbur, and G.J. Dobeck. 2000. Underwater target detection using multichannel subband adaptive filtering and high-order correlation schemes. *IEEE Journal of Oceanic Engineering*.
- [28] Tao Zhang, Xiaqin Li, Yi Zhu, Song Chen, Yu Cheng, and Jingyan Song. dec. 2009. Formation and obstacle avoidance in the unknown environment of multi-robot system. pages 729 –734, *2009 IEEE International*

Figure 1. Example of APF action.

(a)

(b)

Figure 2. (a) First phase of mission planner: the area is discretized as a Grid-Based Map and AUVs reach waypoints;
 (b) Second phase of mission planner: the AUVs are guided towards the target with a continuous control law.

Figure 3. Schematic representation of the first phase of the algorithm.

Figure 4. Simplified model of the AOF forces evaluated from both the GM and the LM, where the yellow centered square represents the AUV.

Figure 5. Schematic representation of the interaction force between robots.

Figure 6. Force trend and distance error from the modified Van der Pol oscillator.

Figure 7. Waypoints guidance by LOS: an example.

Figure 8. Result of a complete mission of 3 AUVs with a fixed target. $\square$ starting position, $\bullet$ position when there is the switching between I and II phase, $*$ final position of the AUVs.

Figure 9. Formation of three AUVs avoiding an obstacle.

Figure 10. Simulation results with mobile target: (a) tracking of a target with an elliptical path; (b) distance error between the first robot and the target.

Figure 11. Simulation results with mobile target: (a) tracking of a target with a sinusoidal path; (b) distance error between the first robot and the target.